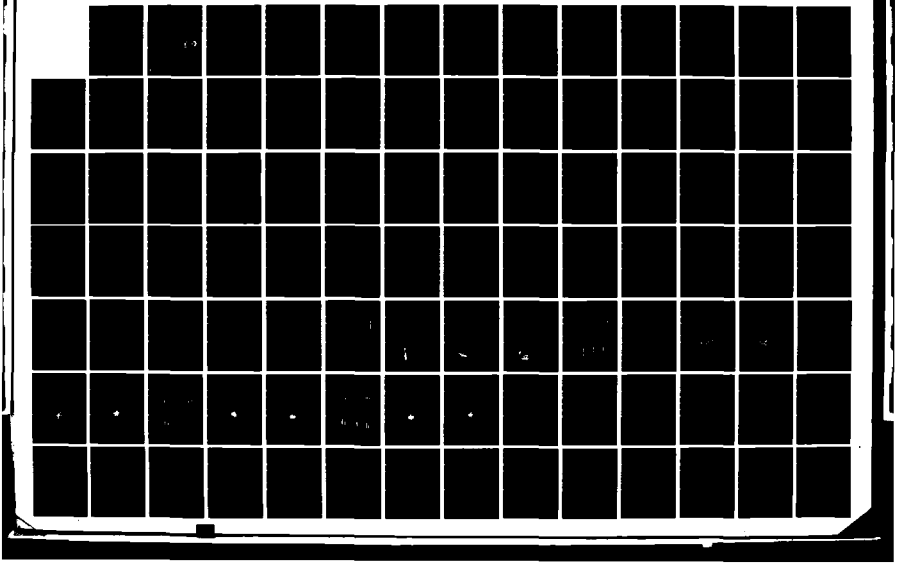
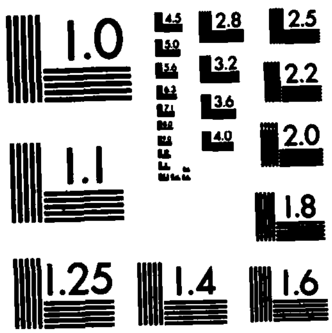
AD-A123 402 MACHINE CLASSIFICATION OF CLOUD PARTICLE TYPES(U) ADAPT 1/2
SERVICE CORP READING MA H E HUNTER AUG 82 ADAPT-82-4
AFGL-TR-82-0298 F19628-81-C-0047

UNCLASSIFIED F/G 4/2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

(12)

AFGL-TR-82-0298

MACHINE CLASSIFICATION OF CLOUD PARTICLE TYPES

Herbert E. Hunter

ADAPT Service Corporation
P.O. Box 58
Reading, MA 01867

Final Report
January 1981 - July 1982

August 1982

Approved for public release; distribution unlimited

AIR FORCE GEOPHYSICS LABORATORY
AIR FORCE SYSTEMS COMMAND
UNITED STATES AIR FORCE
HANSCOM AFB, MASSACHUSETTS 01731

MIL-STD-847A
31 January 1973

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFGL-TR-82-0298	2. GOVT ACCESSION NO. AD-A123 402	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) MACHINE CLASSIFICATION OF CLOUD PARTICLE TYPES		5. TYPE OF REPORT & PERIOD COVERED Final Report Jan 1981-July 1982
		6. PERFORMING ORG. REPORT NUMBER ADAPT 82-4
7. AUTHOR(s) HERBERT E. HUNTER		8. CONTRACT OR GRANT NUMBER(s) F19628-81-C-0047
9. PERFORMING ORGANIZATION NAME AND ADDRESS ADAPT Service Corporation P.O. Box 58 Reading, Mass. 01867		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 62101F 667012BE
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Geopysics Laboratory Hanscom AFB, Massachusetts 01731 Monitor/Rosemary M. Dyer/LYC		12. REPORT DATE August 1982 13. NUMBER OF PAGES 114
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Cloud Physics, Ice Crystals, Snow growth, Hydrometeor, Pattern Recognition, Eigenvectors, Empirical Orthogonal Functions, Screening Classifiers

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Classification algorithms have been developed to separate cloud particles into: 1) dendrites, 2) needles, 3) columns, 4) plates, 5) streakers, and 6) a miscellaneous or an unclassifiable class. These algorithms have been incorporated in schema which when applied to shadow graph images produced by the Knollenberg laser scanning device

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

have demonstrated a capability to classify more accurately than human classifiers and a relative insensitivity to particle orientation.

The training data consisted of a specially selected set of observations obtained on four flights. The algorithms were tested against arbitrarily selected segments from two additional flights. The algorithms were developed using the ADAPT Service Corporation's eigenvector or empirical orthogonal function (EOF) technique to objectively define the features and the ADAPT independent eigenscreening algorithm development program to relate these features to the particle types.

Analysis of the performance suggests that considerable variation is to be expected based on the set to set variation of particle types and distribution between real data sets. The classification schema have been developed to allow the user to change key parameters to compensate for this variation. The use of the confusion matrix to select the value of these parameters is illustrated.

Human and machine classification of these particles was compared. It was found that there is considerable disagreement between classifications made by two different observers trained by the same person as well as considerable disagreement between classifications made by the same observer at different times. A team method was introduced utilizing two human classifiers and a preliminary machine classification to attempt to minimize this effect in creating the training data set. It was concluded that the machine classification developed was greatly superior to manual classification for day to day identification of these particles because the machine classifiers were much faster, less costly, did not suffer from fatigue and were usually more accurate, (i.e. in better agreement with the trainer) then human classifications.

ACKNOWLEDGEMENT

The author would like to express his appreciation to Rosemary Dyer andMorton Glassof Air Force Geophysics Laboratory for the many useful discussions and for the great amount of effort put into classifying the training particle types. The author would also like to acknowledge the financial support of the Air Force Geophysics Laboratory under contract No. F19628-81-C-0047 which made this study possible.

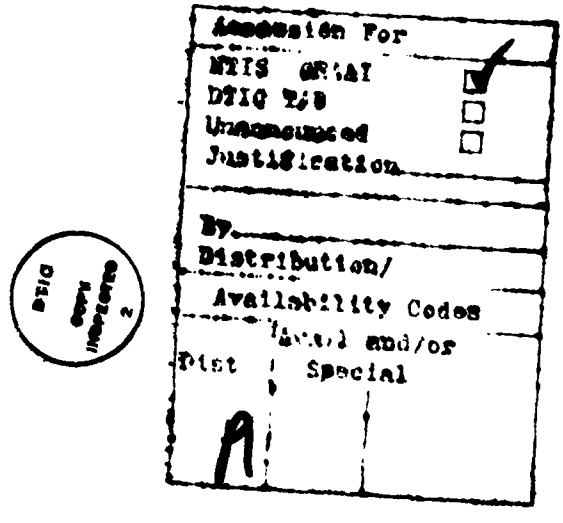


TABLE OF CONTENTS

1.0 INTRODUCTION

The problem of identifying ice particle types occurring in clouds based on shadow graph images have recently become important to cloud physics studies. The effects of clouds on microwave sensing in weapon systems, satellite imagery, and the propagation of millimeter waves are often functions of the shapes of cloud particles as well as their concentration and size distribution. Dyer and Barnes[1] have presented a survey discussion of the areas of applications, measurement techniques and characteristics of ice crystals which can be found in clouds. Additional discussion of applications and measurement of cloud particle shapes can be found in References 1 and 2.

The problem of examining the two dimensional images, identifying the particle type and calculating the statistics of particle types occurring during the data collection run is one which is particularly well suited for automated machine classification. This is because of the large quantity of data which is obtained and the fact that the manual classification of particles very rapidly becomes fatiguing and would require a very large staff of technicians to classify even a relatively modest set of data. In addition to this, we have shown (see Section 3.0) that the particle types are sufficiently poorly defined that there is significant disagreement among classifications made by different persons, even when those persons are aware of the problem and trained by the same person. We have also found significant disagreement between classifications made by the same person at different times.

Despite this great need with the exception of the classifiers presented in References (1) and (2), the development of automated machine classifiers for recognition of cloud particle types have not been reported in the literature. Although the machine classifiers which have been presented in this paper apply to the same data as those which are described in Reference (1) and (2), there are several significant differences between the classifers developed here and those reported in References (1) and (2) which should significantly improve their performance.

---

(1) Rosemary M. Dyer and Arnold A. Barnes, Jr., "The Microphysics of Ice Clouds-A Survey", Air Force Surveys in Geophysics #411, AFGL-TR-79-0103, NTIS AD A07702D, 8 May 1979.

These are:

1) The present classifiers were developed using real data as training data.

2) The present classifiers use the unique capability of the ADAPT family of empirical classification development programs to objectively define and select features for large data sets as opposed to the usual intuitive definition of features followed by an objective evaluation of a limited number of features.

3) The present classifiers are based on variations of the Fisher classifier rather than the maximum likelihood/ Bayes family of classifiers.

4) The present algorithms are incorporated in schema which permit the adjustment of thresholds to account for special needs of the user or characteristics of the particular data set.

Section 2 of this report summarizes the development and performance of the machine classifier. Section 3 summarizes the comparison of human and machine performance for making these classifications. Sections 2 and 3 have served as the basis for two separate journal articles to be submitted on these two subjects, respectively.

Section 4 of this report briefly summarizes the use of the computer programs which have been developed to prepare the data and implement the classification procedures. The data preparation program is a relatively minor modification from the users viewpoint of the Air Force Geophysics Labs Program KN2UTIL. The program for implementing the classification algorithms takes the output of the modified program KN2UTIL and processes it through the schema resulting in a printout of the identification of the particles. Sections 5 and 6, Analysis of Eigenvector Transformation and Analysis of Relative Importance Vectors provide detail insight into some of the characteristics of both the eigenvector transformation and the algorithms. This information is of interest to those who wish to understand the mechanisms of the algorithms and more about the eigenvector approach used to develop the algorithms, however, it is not necessary to the understanding of what the algorithms will do and how to use the algorithms. The final Section 7.0, summarizes the conclusions which we have reached as a result of this study.

2.0 DEVELOPMENT OF 2-D HYDROMETEOR MACHINE CLASSIFIER FROM OBSERVED DATA

2.1 Definition of Data Vectors

Data Source

The data used in this study were obtained from the two dimensional Knollenberg laser scanning device. These devices and their calibration are described extensively in the literature by Knollenberg (4), (5), (6) and by Heynsfield and Knollenberg (7), Heynsfield (8), and Cunningham (9). Briefly, the system consists of a laser beam luminating a line of photo diodes. As particles fall through the viewing volume, it includes some of the diodes, the number and location of which are determined by the particle size and shape. A rapid scanning system records the diodes included per unit forward motion of the aircraft. This forward motion equals the minimum grid size; and hence, the smallest size particle measurement by the device (25μ). Figure 1 is an usually clear example of an unusually pure set of dendrites recorded by the two dimensional probe.

Preprocessing

The initial preprocessing performed is to reject artifacts and trivial cases. The most common of these to be rejected are:

1) particles less than three diodes in length. Particles smaller than three diodes do not have sufficient information for classification and may be treated as spheres for many of the applications.

2) Images containing more than one particle. Multiple particles are rejected primarily to make the classification problem more tractable. Physical arguments can also be made that when multiple particles are present they are often pieces of a single particle that is breaking up.

3) Particles which were entering broadside were also rejected to make the classification problem more tractable. These particles could be handled by rotating them 90 degrees and then processing them through the classifier. The programs developed are capable of this, although to date, these particles have represented a relatively small percentage of the total particles and this has not been necessary.

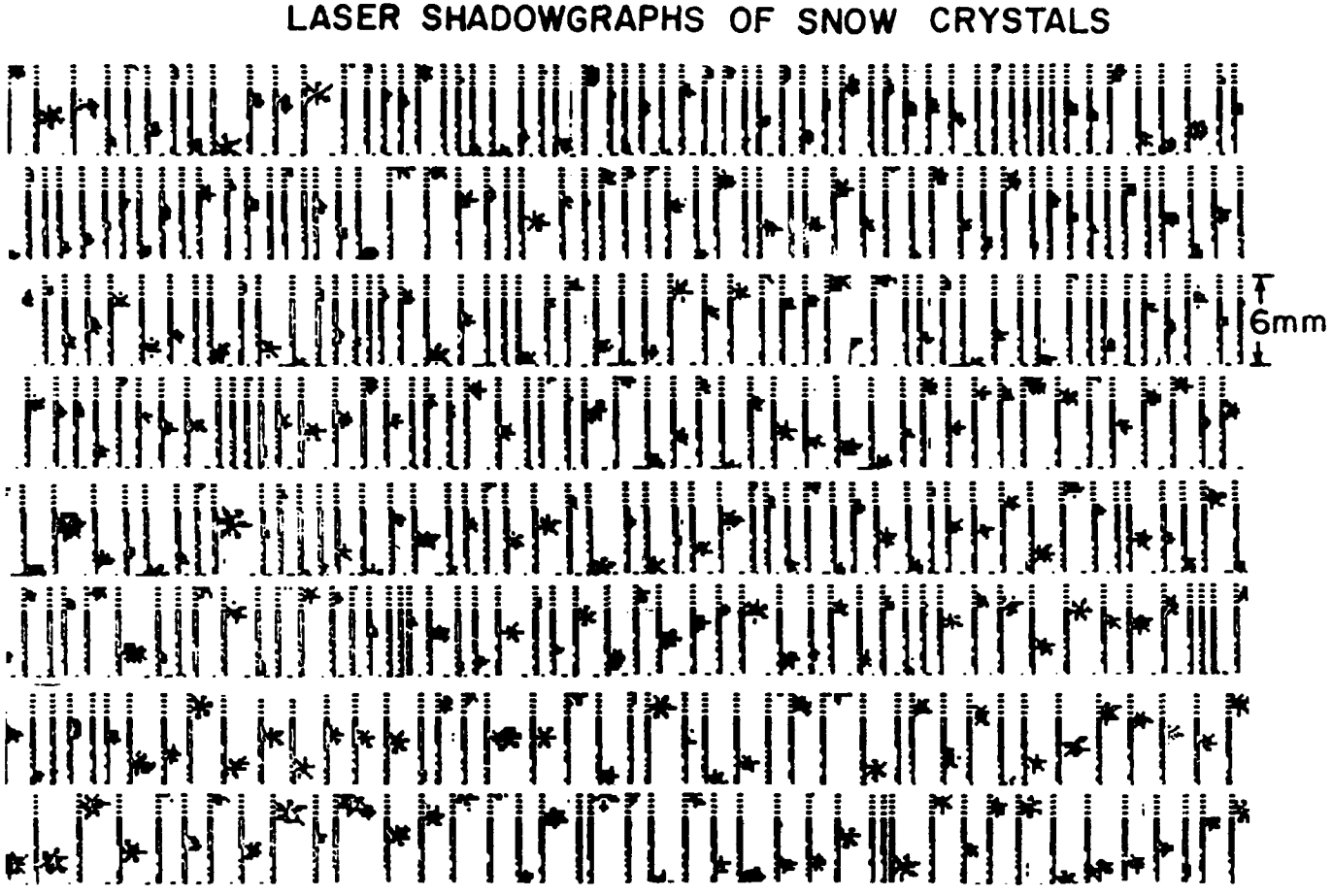


Figure 1 Dendritic Crystals Recorded by the PMS 2-D Precipitation Probe

Unlike the work presented in References (2) and (3), streakers (usually water shedding across the up stream edge of the probe and across the aperture) have not been rejected in the preprocessing stage but have been retained as a particle class to be rejected as part of the classification procedure. This has been done since the processing required to identify streakers is more like the classification of the particles than like the remainder of the preprocessing in terms of the procedures used and can be done more efficiently at the latter time. An optional preprocessing is available as part of the classification procedure to rotate all particles until their ratio of the maximum width to length is minimized. This option was used primarily to investigate the sensitivity of the algorithms to rotation and will be discussed later in the paper.

Construction of Data Vectors

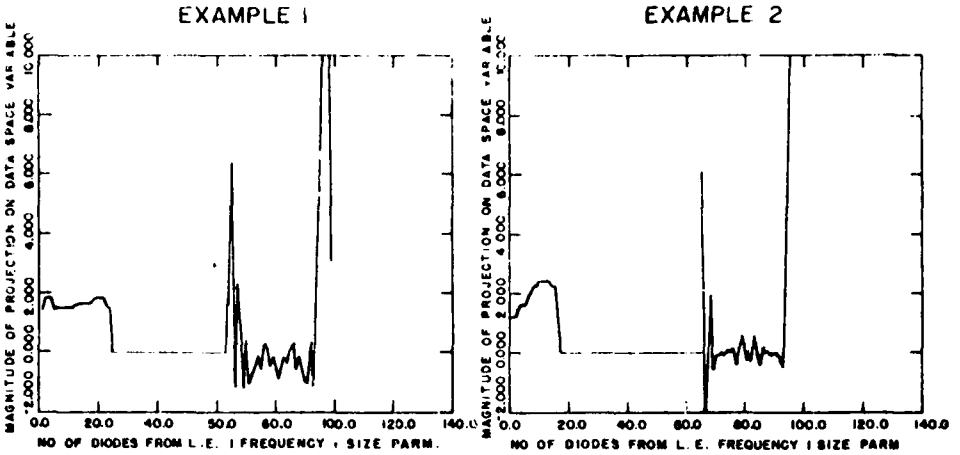
The second step in the data preparation was to construc a linear data vector from the two dimensional binary arrays of occluded and exposed diodes.

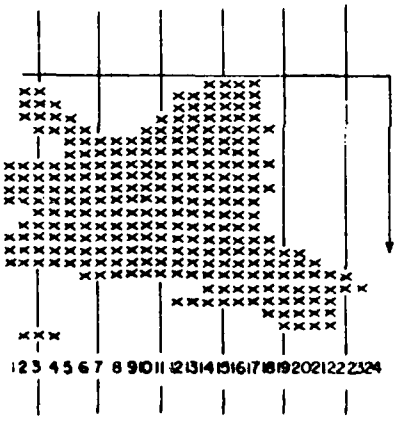
Figure 2 illustrates how the 2-D spectrometer data was converted into a data vector. This figure illustrates this procedure for two different particles used in this study. The upper portion of the figure is the data vector corresponding to the shadow image shown in the lower portion of the figure. The shadow of the particle is created by the occlusion of diodes in a 32 by N array as the particle passes. Each of the X's in the lower figure indicates the location of an occluded diode. The shadow graphs of the particles were relocated so that the origin could be taken in the upper left hand corner of these figures and at least one occluded diode would lie on the horizontal axis.

To create a single numerical value for each row, each occluded diode was treated as a binary bit turned on and each of the diodes which was not included was treated as a zero binary bit. This bit structure at each of the horizontal locations was then interpreted as a binary number. For the purposes of this study, the length of the particle, N, was arbitrarily limited to a maximum of 64 rows of diodes. Thus, the first 64 numbers in the upper figure or data vector are simply the natural logarithm of the integer resulting from the bit structure of the corresponding row of diodes normalized to unity square magnitude. The next 30 numbers are the frequency spectra obtained when that 64 point bit structure was processed through a fast Fourier transform. To eliminate the DC term, the first two bins of the frequency spectra were deleted

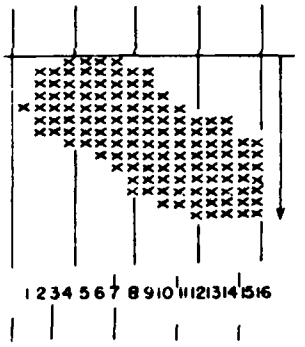
FIG. 2

ILLUSTRATION OF RELATION BETWEEN SHADOW IMAGE AND DATA VECTOR





PARTICLE IMAGE CORRESPONDING TO COMPONENTS 1-24 OF THE DATA VECTOR



PARTICLE IMAGE CORRESPONDING TO COMPONENTS 1-16 OF THE DATA VECTOR

resulting in a total of 30 numbers. The 30 frequency bins were also normalized to unity square magnitude. The 95th and 96th values in the data history were the square magnitude of the time history and the square magnitude of the frequency histories which were used to normalize these two portions of the history. Points 97 and 98 were the Feret length and width, respectively. The 99th point was the index value at which the particle had its maximum width.

2.2 Definition of Truth Data Sets

A total of 2,104 particles images obtained on three data gathering flights were used as training and verification data for this study. This data was divided into three different sets of data. The training set consisted of 403 patterns obtained on a single flight. There were also two verification sets taken from two different flights. One verification set consisted of 1500 particles and the second of 201 particles. The 201 case set was used to evaluate both manual classification and competitive classification schemes and will be called the "manual reference set" through the remainder of this paper.

Manual classification were prepared for all three data sets. Except for the 1500 case proof test set, at least two different manual classifications were made for each of these data sets. For each of these three data sets, the manual classification arrived at by the team effort and defined as the correct classification was available. For the 403 training cases, there was also the initial manual identification used to derive the first exploratory algorithms. We shall refer to this manual classification as the original training classification. Approximately a dozen different manual classifications were made on the manual reference set and are reported in detail in Reference 3.

The "correct" identifications for this study were based on a team effort which was used to minimize the problem of consistency of manual classification. The problem of the consistency in the definition of the images first become apparent in the task of defining the truth data for developing the automated classifier. It was found that for a number of particles considerable disagreement was occurring between qualified meteorologists. This problem is discussed in detail in Reference 3. The initial truth data was picked by Rosemary Dyer after detail discussion with the other two authors. After noting the substantial disagreement with the development of an exploratory set of algorithms using the initial set of truth data at the same time carry out a study to determine the severity of this problem.

The training set consisted of 403 patterns which have been divided into four non-trivial classes, one trivial class and a class of unclassifiable shadows. The trivial class containing only three member was a streaker class for which there are a number of strong characteristics which are reflected in the eigenvector expansion and can easily be identified by machine and thus the algorithms for this class were developed by inspection of the projection on the eigenvector space. There is no disagreement among the human classifiers as to correct identification of the members of this class.

A second data set completely different from the 403 training cases consisting of 201 cases was selected as a set to be analyzed by a number of different observers to allow a study of the consistency of human observations. After completing their individual analysis of the 201 case comparison set, the two authors located at AFGL undertook a joint examination of a total of 2,104 images that included both the 403 training images, the 201 manual reference images and an additional 1500 images. These images were arranged in an order so that the authors were not aware of the places in which the original training or the 201 case manual reference set occurred. A period of time not exceeding 1 hour was set aside each day to work on this task to prevent fatigue which was known to cause a change in the identification. When these images were supplied to the two AFGL authors, the machine classifications using the algorithms based on the original 403 case training set were also furnished to improve their consistency with the original training set. The results of this careful team analysis were then taken as the "correct" or truth data for this study. It should be remarked that a different set of people or even the same authors performing a different time would have somewhat different results, however, this was felt to be the best result which could be obtained and will provide a good basis for comparison of consistency. It is also the best available basis for developing the machine classification algorithms.

2.3 Design of Machine Classifiers

The classifiers developed were Fisher classifiers which were developed using the data vectors derived from the shadow graphs. These signatures were preprocessed through an eigenvector transformation and then the Fisher classifier developed using an independent eigenscreening scheme. The general concept of the ADAPT Service Corporation's independent eigenscreening approach to deriving classifiers such as the Fisher

classifier has been described in Reference (10) and in much greater detail by Hunter[2]. This report includes an appendix illustrating the difficulty with conventional iterative eigenvector technique which are overcome by the ADAPT eigenvector programs. The ADAPT approach may be briefly summarized as follows. The training data vectors to be used to develop the classification algorithm are first used to develop the transformation to their eigenvector space (i.e. the optimum empirical orthogonal functions, E.O.F.). This transformation is then used to transform this data to the eigenvector space. A screening procedure is then used to develop the Fisher classifier where the screening is performed on the projections on each of the eigendirections (E.O.F.'s) as opposed to screening on the original variables. The screening also differs from conventional "forward sequential" selection in that the screening decision to keep or reject a given eigendirection in the classifier is based on an unbiased performance estimate using the modified "one-out" method of Reference 11. Thus, the approach from conventional screening primarily in two ways: 1) the use of the components in eigenvector space as independent variables, and 2) the use of an unbiased performance estimate as opposed to the normal procedure of using dependent or biased test results to make the screening decisions.

These procedures were used to develop five separate algorithms for classifying dendrites, needles, columns and plates. A fifth algorithm for classification of streakers was developed by examining the projection of the data on the first two eigendirections.

Thus, the machine classification scheme was built around these five algorithms each of which was of the same mathematical form. That is, each algorithm consisted of a vector onto which all of the data was projected. This vector was selected to maximize the ratio of the interclass dispersion to the intra-class dispersion using the Fisher criteria for the first four algorithms and by examination of the projection of the data on the first two eigendirections for the fifth algorithm. These

---

(2) Hunter, H.E.; "Final Report, NCSC Scale Model Classification Potential", Contract N61331-79-C-0038, ADAPT Report 80-4, DTIC #AD-B062 5576, Dec 1980.

five algorithms were utilized to identify the most likely class by calculating the five likelihood ratios based on the detection statistics obtained by applying each of these five algorithms. The algorithm displaying the highest likelihood ratio was used to define the class to which the shadow graph belonged.

Definition of Features

The objective features which are used for this analysis are the E.O.F.'s or the eigenvectors of the variance-covariance matrix associated with the 403 training data vectors as described in Section 2. These features have the advantage that they are orthogonal and are complete; that is, all of the features have been constructed which are required to completely define the variation in the data. Recently there has been considerable discussion in the literature (12), (13) of the question of the significance of the higher order eigenvectors. The techniques which are suggested for demonstrating this significance are clearly stated as being sufficient but not necessary. Our experience with several hundred eigenvector problems has shown that these criteria are very conservative and often result in the rejection of the most significant data for a difficult problem. We have found the most effective measure of significance is the usefulness of the eigenvector (that would be used for any other feature) *in either the regression or* classification algorithm. A possible reason this approach is not used more in the literature is that screening decisions are usually based on dependent (i.e. biased) test results instead of independent tests. The ADAPT techniques base these decisions on independent tests!

One reason the method of Reference (12) is overly restrictive is that if we are considering a higher order eigendirection, (for example, the Nth) the random vectors model the entire reconstruction rather than the portion using only greater than the Nth terms. A less conservative approach would be to reconstruct the data vectors using the first N eigenvectors, substract this reconstruction from each of the data vectors and then construct the eigenvector expansion from the remaining portion of data vectors and compare the eigenvalues associated with this expansion with that generated from a set of corresponding random vectors. The computation involved in this would be far greater than that of the method given in Reference (12) and it still would yield a sufficient but not necessary condition. For the present study, we have found that significant information can be found up to and including the twentieth eigenvector.

A second problem associated with optimal empirical functions or eigenvectors is that, in general, the iterative techniques for finding eigenvectors which appear in many of the statistical packages do not produce correct eigenvectors for the higher order eigenvectors when the data set involved is large and/or noisy. This may be easily demonstrated by inserting relatively simple sets of vectors into these and noting it is possible to obtain more eigenvectors having positive eigenvalues from these procedures than would be possible with a correct derivation. This is illustrated in the appendix to Hunter[2].

Performance of Individual Algorithms

Three types of classification algorithms were derived and later combined into a number of different schema for performing the required separation into the six classes. The first type of classification algorithm would be more properly called a detection algorithm. This type of algorithm was developed to detect a given class versus all other classes. For this type of algorithm, Class 1 was made up of the class to be detected. Class 2 was made up of all of the other classes of interest and the miscellaneous and streakers were omitted from the algorithm development.

The second type classification algorithm developed was the classification of one class versus the second class. In this case, only two classes of interest were used in the algorithm development with Class 1 being one of the classes and Class 2 the other class. All other classes as well as the miscellaneous and streakers were omitted from the algorithm development.

The third group of classification algorithms is actually a special case of the first two. In this case, two classes were separated from the other two classes. For example, Class 1 could be dendrites and plates and Class 2 would be needles and columns. Again, miscellaneous and streakers were omitted.

Table 1 presents a summary of performance of all of these algorithms. The performance given is in terms the Fisher parameter and an equivalent probability of error. The Fisher parameter is the parameter which is minimized by the Fisher classifier and is simply the ratio of the sum of the standard deviations of the two classes used to develop the algorithm (the within class variation), to the distance between the means of these two classes (the between class variation) when

TABLE 1 - PERFORMANCE OF FISHER CLASSIFICATION ALGORITHMS
USED FOR DESIGN OF CLASSIFICATION SCHEMA

ALGORITHM	DEFINITION OF CLASSES		PERFORMANCE	
	CLASS I	CLASS II	FISHER PAR	EQUIV PE
One (1) Class Detectors				
Dendrite Detector	Den.	Streakers & Misc	0.783	0.095
Needle Detector	Ned.	"	0.647	0.06
Column Detector	Col.	"	1.59	0.27
Plate Detector	Plate	"	0.80	0.10
Two (2) Class Detectors				
Column and Needle Detector	Col&Ned	"	0.79	0.10
Column and Plate Detector	Col&Plate	"	1.30	0.16
Column and Dendrite Detector	Col&Den	"	2.00	0.30
Classification Algorithms				
Dendrite vs Needle Class.	Den	All but Den&Ned	0.424	0.02
Dendrite vs Column Class.	Den	All but Den&Col	0.685	0.07
Dendrite vs Plate Class.	Den	All but Den& Plate	0.966	0.15
Needle vs Column Class.	Needle	All but Ned&Col	0.891	0.13
Needle vs Plate Class.	Needle	All but Ned&Plate	0.535	0.03
Plate vs Column Class.	Plate	All but Plate&Col	0.784	0.095

the classes are projected onto the Fisher derived optimal separation direction. The equivalent probability of error is included because the non-linearity of the Fisher parameter makes it difficult to grasp its physical meaning. We define the equivalent probability error as the probability of making any error in the classification which would be associated with a particular Fisher parameter for a classification where the standard deviation of Class 1 equals the standard deviation of Class 2. This definition has the advantage that it allows a unique relationship between a probability of error and the Fisher parameter. In general, the probability of error will depend on both the Fisher parameter and the relative sizes of the standard deviation of the two classes and the threshold selected. All of the performances given in Table 1 are based on the modification of the Lachenbach (11) "one-out" method. This modification is to use groups of observations rather than single observations in the procedure.

In examining Table 1, it is useful to realize that human classifiers tend to have probability of errors ranging from 0.25 to 0.65 as compared to the correct classification and from 0.25 to 0.40 in terms of agreement between the same person performing the classification at different times. These results are discussed in detail in Section 3.0.

The first column of Table 1 defines the algorithm and associates it with one of the three groups of algorithms previously described. The next two columns of Table 1 define the members of the three classes used in the development of the Fisher algorithm. Where Class 1 is the first class in the development of the Fisher algorithm, Class 2 is the second class in the development of the Fisher algorithm and Class 3 are those classes which were omitted from the development of the algorithm. The fourth and fifth columns give the Fisher parameter and equivalent probability of error, respectively.

2.4 Definition of Classification Schema

The information presented in Table 1 was used as a basis for selecting two schema for combining these algorithms into a classification decision in an automated manner. Figure 3 presents a diagram of the two schema which were developed for this study.

1-Step Classification Schema

The one step schema shown at the top of this figure represents the simplest approach to using the algorithms to make the decision between which of the six classes; that is,

1) dendrites, 2) needles, 3) columns, 4) plates, 5) streakers, and 6) miscellaneous or unclassifiable each particle belonged. It consists simply of processing an unknown particle data vector through the five detection algorithms and comparing the projection of the particle on each of the five Fisher directions with the statistics of the projection of the training data on these directions. Based on this projection and the statistics of the training data, one can calculate a likelihood ratio that the particle belongs to the class for each of the five algorithms. The likelihood ratios are then compared with a threshold and if none of the likelihood ratios exceed this threshold, the particle is considered unclassifiable (i.e. a member of the miscellaneous class). If one or more of the likelihood ratios exceeds the threshold, the particle is associated with the detection algorithm for which the likelihood ratio is the greatest.

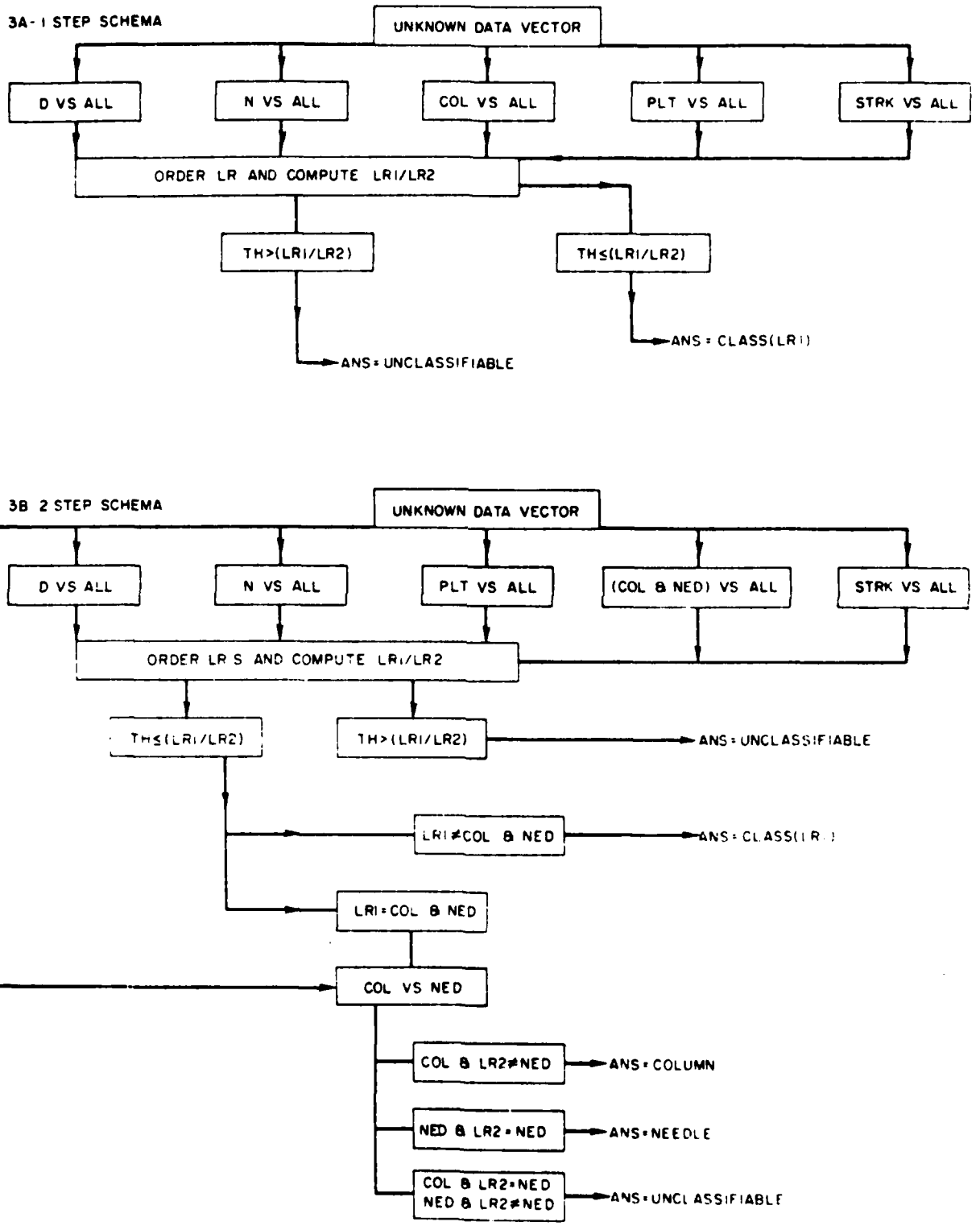
This schema is closely related to a maximum liklihood ratio approach and, in fact, will reduce to a maximum likelihood ratio for those special cases for which the Fisher classifier reduces to the maximum likelihood ratio classifier. However, this approach results in a significant reduction in the computation required to apply the algorithm. It also provides for considerable additional flexibility since both the statistics of the training data and the characteristics of the individual algorithms can be modified by an appropriate input. Thus, the algorithms may be adjusted or tailored to specific apriori information without a rederivation of the entire algorithm.

2-Step Classification Schema

Figure 3B is an example of a modification of the simple schema shown in Figure 3A which is possible by adding a single additional algorithm and replacing one of the algorithms to overcome a particular difficulty with the algorithms. Examination of Table 1 shows that all four of the most important algorithms used in the schema of Figure 3A have equivalent probability of errors of less than 10% except for the column detector algorithm which has equivalent probability of error of 27%. Further examination of Table 3 suggests that if the columns were detected using a two step procedure of first detecting combinations of columns and needles and then separating the columns from needles with the column versus needle classification algorithm, this probability of error might be reduced to approximately 22%. Thus, the schema shown in Figure 3B was developed to implement this approach.

FIGURE 3

SCHEMA FLOW DIAGRAMS



Initially, the two step schema shown in Figure 3B is essentially identical to the one step shown in Figure 3A with the exception that the column detector has been replaced with the column plus needle detector algorithm. If the likelihood ratio is less than the threshold or if the maximum likelihood ratio is associated with the dendrites, needles, plates or streakers, the procedure and results are identical to the one step schema shown in Figure 3A. However, if the maximum likelihood ratio is associated with the columns and needles class, the data vector associated with the particle is also processed through the columns versus needles classification algorithm. The results of this classification are then compared with the second largest likelihood ratio and a decision is reached as to whether the particle should be called a column, a needle, or a miscellaneous particle.

2.5 Performance of Classification Schema

Confusion Matrix Performance Measure

The performance of the raw algorithms was summarized in Table 1 in terms of the Fisher parameter and its associated equivalent probability of error. However, the performance of interest is that of the combination of these algorithms for the overall task of identifying the particle type associated with any data vector. Although the Fisher parameter is an excellent measure of performance of the individual algorithm, it is not suitable for evaluating this multiple class schema or combination of algorithms. One of the more efficient measures of performance for the combination of algorithms is the confusion matrix. This is a matrix which has the class as both of its axis. One axis in our paper, the vertical axis, represents the actual class of the particle the other axis in our paper, the horizontal axis is the class which the particle was identified as. Thus, the diagonal of this matrix is simply the number of correct classifications for each algorithm.

Figure 4A presents this confusion matrix for the application of the one step algorithm to the 403 training cases using the group out modification of the Lachenbrach (11) one out method. Examining the first row of this matrix, we see that from the first class which had a total of 63 members, 46 of these were correctly identified as dendrites, none of them were incorrectly identified as needles, four of them were incorrectly identified as columns, 11 of them were incorrectly identified as plates, none were incorrectly identified as streakers and two of them were incorrectly identified as miscellaneous. Similarly, one can get the exact performance for each of the other classes by examination of this matrix. It is somewhat more useful to normalize the confusion matrix by dividing each row by the

FIGURE - 4 CONFUSION MATRIX FOR GROUP-OUT TESTING OF THE TRAINING SET OF CLOUD PARTICLES

A) CONFUSION MATRIX BASED ON COUNTS

ACTUAL CLASS	CALLED CLASS 1	2	3	4	5	6
1	46.00	0.0	4.00	11.00	0.0	2.00
2	1.00	90.00	11.00	0.0	0.0	3.00
3	6.00	8.00	62.00	3.00	0.0	10.00
4	8.00	5.00	9.00	90.00	0.0	8.00
5	1.00	0.0	0.0	0.0	2.00	0.0
6	1.00	10.00	6.00	4.00	0.0	2.00

B) NORMALIZED CONFUSION MATRIX

ACTUAL CLASS	CALLED CLASS 1	2	3	4	5	6
1	0.73	0.0	0.06	0.17	0.0	0.03
2	0.01	0.86	0.10	0.0	0.0	0.03
3	0.07	0.09	0.70	0.03	0.0	0.11
4	0.0,	0.04	0.07	0.75	0.0	0.07
5	0.33	0.0	0.0	0.0	0.67	0.0
6	0.04	0.43	0.26	0.17	0.0	0.09

FIGURE - 5 CONFUSION MATRIX FOR THE MANUAL REFERENCE SET OF CLOUD PARTICLES

CONFUSION MATRIX BASED ON COUNTS

ACTUAL CLASS	CALLED CLASS 1	2	3	4	5	6
1	37.00	0.0	1.00	4.00	0.0	0.0
2	3.00	16.00	5.00	2.00	0.0	1.00
3	46.00	2.00	32.00	10.00	0.0	3.00
4	7.00	0.0	3.00	13.00	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0
6	1.00	4.00	2.00	7.00	0.0	2.00

NORMALIZED CONFUSION MATRIX

ACTUAL CLASS	CALLED CLASS 1	2	3	4	5	6
1	0.88	0.0	0.02	0.10	0.0	0.0
2	0.11	0.59	0.19	0.07	0.0	0.04
3	0.49	0.02	0.34	0.11	0.0	0.03
4	0.30	0.0	0.13	0.57	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0
6	0.06	0.25	0.13	0.44	0.0	0.13

actual number of members of that class, thus, each value becomes the decimal fraction of identifications. This we call the normalized confusion matrix and is shown in Figure 4B for the reference one step algorithm corresponding to Fisher 4A. Thus, once again examining the first row, we see that 73% of the dendrites were correctly identified as dendrites and 6% of the dendrite incorrectly identified as columns, 17% incorrectly identified as plates and 3% incorrectly identified as miscellaneous.

The confusion matrix provides us a complete analysis of the performance of the algorithm. However, it has the disadvantage that it is rather cumbersome to compare a number of algorithms using the confusion matrix. One relatively simple reduction of this cumbersomeness is to deal only with the diagonal of the confusion matrix. Referring to Figure 4B, we see that this confusion matrix has a diagonal having the values .73, .86, .70, .75, .67 and .09. This tells us that the reference one step algorithm correctly identified 73% of the dendrites, 86% of the needles, 70% of the columns, 75% of the plates, two-thirds of the streakers and 9% of the miscellaneous. Thus, the diagonal of the confusion matrix can by itself give us considerable information regarding the performance of the classification schema. It still defines the performance in terms of probability of correct identification (or probability of error) for each of the classes. What we are missing is how the errors are distributed amongst the remaining classes. Finally, if one were to construct the weighted average of the normalized diagonal where each of the diagonal values were weighted according to the total membership of the corresponding class in the set and the resulted weighting average expressed as a percent, one would have the percent of correct identifications, which will be used as the primary comparison method in Section 3. Thus, one may view the confusion matrix as a generalization of % correct for a binary class problem to a multiple class problem.

Comparison of Classification Performances for Schema and Modifications

In evaluating the performance of these classification schema, the reader is reminded that the most significant performances are those associated with Classes 1 through 4 and that the performance of manual classification tends to range between 35 and 75 in terms of the weighted average of the diagonal. The reason for the lesser importance of classes 5 and 6 are: 1) class 5 only has three members and thus makes a very small contribution to the weighted average and class 6 also only as 23 members. It should also be noted that very little effort was exerted on developing a highly efficient

streaker (class-5) detection algorithm since apriori procedures are already available which can be used to prescreen the streakers if necessary. The miscellaneous class, class 6, can be eliminated from the procedure by adjustment of the likelihood ratio thresholds. Complete elimination of any membership of this class does not significantly effect the overall performance of the schema.

Performance on Selected Data Set

Table 2 presents a comparion of the performance of a number of variations of one and two step schema on the training set of 403 cases. These performances are based on the group-out test modification of the Lachenbrach (11) one-out method. The first column of Table 2 defines the schema used, i.e. whether it is a one step or two step and if applicable a descriptive name. The next three columns of various thresholds which can be set to modify the schema. Results are given for three of the more significant thresholds. These are the threshold on the likelihood ratio (LR), the threshold on the dendrite detection algorithm and the threshold on the second step algorithm for separating needles and columns. The fifth column of Table 2 indicates whether the schema was applied to rotated or unrotated data. The last six columns give the diagonal values of the confusion matrix associated with each of the six classes. The first case shown in Table 2 is the nominal one step algorithm for which the confusion matrix was given in Figure 4. For this schema, the likelihood threshold was set at an absolute value of one thus it is indicated as 1.0A. The dendrites threshold was set at the 0 value which corresponds to the minimum total errors using the Anderson-Bahadur(14) approach and since this was a one step algorithm, the second step threshold is not applicable. All of the data discussed to this point in the paper has been unrotated data. Examination of the last six columns for the 1-step nominal schema shows the diagonal values which were given in Figure 4.

This training set of data differed considerably from the two test sets which will be discussed later. The training set was a set of data which was picked because it contained unusually good examples for real data and an unusual variety of cases. Thus, the training set could not be considered a typical data set but rather an unusually pure set.

The nominal schema was modified by varying the threshold for the likelihood ratio through a number of different values and the best value determined experimentally. The characteristics and performance of this algorithm are entered in the second row of Table 2. It was found that a slight improvement was

TABLE 2 - COMPARISON OF SCHEMA PERFORMANCE USING CONFUSION MATRIX DIAGONAL ON 403 CAS TRAINING SET

	SCHEMA	THRESHOLDS			DATA	DIAGONAL VALUES FOR:					
		LR	DENDRITE	2ND STEP	ROTATED	DEN	NEEDLE	COL.	PLATE	STREAK	MISC
1	1-STEP (NOMINAL)	1.0A	0	-	NO	0.73	0.86	0.70	0.75	0.67	09
2	1-STEP (BEST)	1.05	0	-	NO	0.73	0.89	0.72	0.80	0.33	0
3	2-STEP (BEST)	1.05	0	-7.5	NO	0.73	0.93	0.71	0.80	0.33	0
4	2-STEP	1.05	0	-6.5	NO	0.73	0.98	0.61	0.80	0.33	0
5	2-STEP	1.05	0	-8.5	NO	0.73	0.91	0.74	0.80	0.33	0
6	1-STEP(BEST/ROTATED)	1.05	0	-	YES	0.70	0.93	0.65	0.78	0.33	0
7	2-STEP(BEST/ROTATED)	1.05	0	-7.5	YES	0.70	0.98	0.64	0.78	0.33	0
8	1-STEP(WEAK DENDRITE)	1.05	-2.2	-	NO	0.51	0.90	0.75	0.85	0.33	0
9	2-SEP (WEAK DENDRITE)	1.05	-2.2	-7.5	NO	0.51	0.94	0.74	0.85	0.33	0

possible by using a relative rather than an absolute threshold on likelihood ratio. In this case, the relative criteria was that a particle was identified as miscellaneous if the largest likelihood ratio exceeded the second largest likelihood ratio by less than five percent. This is indicated by 1.05 in the L/R threshold column. The major reason for this improvement was that the miscellaneous class was reduced to only five particles (none of which were correctly identified!). This relative threshold was introduced in the hope that it would be a better approximation to human determination of unclassifiable. However, the fact that the best use of this threshold was essentially to eliminate the miscellaneous class indicates that neither the absolute or relative thresholds result in a good approximation to what the human does in determining a difficult or unclassifiable particle. This was further verified by use of considerably larger values of the relative ratio where the total number of the miscellaneous class were approximately equal to that of the human classifiers for the training data but the agreement with the human identification as miscellaneous was still extremely poor. This is in agreement with the conclusion of Section 3.0 that different sets of particles were difficult for the machine than for human classifiers.

The information in Rows 3 and 4 compares the performance of the best one step and the best two step algorithms. We see that on this training data set the introduction of the second step has made relatively little difference on the performance of the algorithm. There has been a slight improvement of the identification of the needles and an insignificant decrease in the identification of the columns. The fifth row illustrates the type of effect that can be achieved by modifying the threshold on the second step algorithm by approximately a half of the average standard deviation of Class 1 and Class 2 for this algorithm. The effect has been to significantly increase the performance in detection of needles at the cost of decreasing the detection performance for the columns. The effect of a similar change in this threshold in the opposite direction is shown in row six where we see the effect is in the opposite direction but considerably smaller.

The next two rows , have been included to show the effect of rotating the data. One of the criteria used in defining the preprocessing of the data vector was to minimize the effect of the particles orientation. The first method of doing this was to eliminate the broadside particles from the study since these could be accounted for simply by rotating them 90 degrees as part of the testing procedure. The second approach was to include rotationally insensitive characteristics such

as the Fourier transform in the data vector. To evaluate the effectiveness of these approaches in reducing the sensitivity to rotation, the particles were rotated until they had a minimum ratio of the width to length. Rows 7 and 8 show the performance of both the best one step and the best two step schema on the rotated particles. The conclusion from comparison of these performances with the corresponding performance on the unrotated data is that these algorithms are as hoped relatively insensitive to the particle's orientation. The differences which were observed are quite logical. We note that, in general, the effect of the rotation has been to decrease the performance in the detection of the dendrites columns and plates and to increase the performance for the classification of the needles. The increase of the performance for the classification of the needles is clearly the expected result of rotation. At first, the decrease in performance of the other three classifications might be surprising. Examination of the detail effect of the rotation showed that in addition to making the orientations of all of the particles more similar, the rotation introduced noise due to the finite size of the particles. That is, the edges of the rotated particles tended to be rougher than the edges of the unrotated particles. This introduction of roughness especially for small plates and columns tended to make them look more like dendrites.

The final two rows of Table 2 show the effect of changing the dendrite threshold by one standard deviation of the dendrite training data to weaken the detection of dendrites.

Performance on Manual Reference Data Set

Table 3 shows the comparison of the same schema performances as Table 1 when applied to the 201 case manual reference test set. These are 201 cases which were not part of the training data and which were obtained from typical flights rather than unusually good flights. In fact, postmortem analysis suggests that these flights represented particles which were less well defined then the average case. However, this is the test set which was used to evaluate the manual performances which were discussed in Reference 3 and led to the conclusions that manual classification varied between 35 and 75% correct identification. The format of Table 3 is identical to that of Table 2.

TABLE 3 - COMPARISON OF SCHEMA PERFORMANCE ON USING CONFUSION MATRIX DIAGONAL ON 201 CASE MANUAL REFERENCE SET

	SCHEMA	THRESHOLDS			DATA	DIAGONAL VALUES FOR:					
		LR	DENDRITE	2ND STEP	ROTATED	DEN	NEEDLE	COL	PLATE	STREAK	MISC
1	1-STEP (NOMINAL)	1.0A	0	-	NO	0.88	0.59	0.34	0.57	-	0.13
2	1-STEP(BEST)	1.05	0	-	NO	0.86	0.67	0.42	0.61	-	0
3	2-STEP(BEST)	1.05	0	-7.5	NO	0.86	0.74	0.40	0.61	-	0
4	2-STEP	1.05	0	-6.5	NO	0.86	0.74	0.39	0.61	-	0
5	2-STEP	1.05	0	-8.5	NO	0.86	0.67	0.42	0.61	-	0
6	1-STEP(BEST/ROTATED)	1.05	0	-	YES	0.86	0.70	0.39	0.61	-	0
7	2-STEP(BEST/ROTATED)	1.05	0	-7.5	YES	0.86	0.74	0.40	0.61	-	0
8	1-STEP(WEAK DENDRITE)	1.05	-2.2	-	NO	0.60	0.67	0.58	0.74	-	0
9	2-SEP (WEAK DENDRITE)	1.05	-2.2	-7.5	NO	0.60	0.70	0.59	0.74	-	0

Comparison of Tables 2 and 3 shows that, in general, the performance on the dendrite detection has significantly improved and that the performance on the needles, columns and plates has degraded. This variation in performance is probably typical of variations that can be expected between various data gathering flights depending on the characteristics of the particles. Examination of the particles used on the 201 case manual reference set shows that there are many more particles where it is extremely difficult to decide between needles, columns and plates then was the case in the 403 case training set. The performances suggest the dendrites included in the manual reference set were in general better defined then those in the training set. However, we see that the effect of changing the likelihood ratio threshold criteria is far greater for the manual reference set then it was for the training set and has resulted in a performance significantly improved over the nominal performance especially for the detection of needles, columns and plates.

This example as well as many others which were observed during this study shows that the design of the algorithms for this problem is very sensitive to the particular data set used. For this reason, it seems extremely unlikely that artificially generated particles would lead to algorithms which were useful for the classification of real data. It also suggests that the schema developed must include considerable flexibility to allow the user to adjust the performance of the algorithm to any given data set. This is, of course, possible because the user may apply the algorithms and then manually examine a limited number of the cases to see if the performance is reasonable. By examining the confusion matrix, relative to this determination of correct performance, he can select the thresholds and if necessary the statistics for the algorithms used in the schema. In this way, the algorithms may be optimized for the particular data set and purpose for which the analysis is performed. It would be reasonable to consider these schema "Manually Adaptive". However, the nominal and best one step and two step algorithms will provide significantly better performance against real data sets then can be achieved with manual classifiers. This conclusion is reached by comparing the general level of performances shown in Tables 2 and 3 with the performances ranging from 35% to 75% which are discussed in Reference 3.

It is encouraging to note that despite the sensitivity of the detailed performances to the data set, the effect of rotation on the manual reference data set is also small; in agreement with the effect observed on the training set. Since these data sets are clearly extremes or nearly extremes in real

data, it seems reasonable to assume that we have been successful in at least reducing the effect of rotation to an insignificant problem as compared to other problems associated with the development of these classification schema.

Table 3 illustrates the reason why we have included Rows 9 and 10 in Tables 2 and 3. Although the weakening of the dendrite algorithm appeared to significantly decrease the performance for the 403 case training set, this was not the case for the manual reference set. Although the direction of changes in performance were similar, the reduction in the threshold value for the dendrite algorithm, decreased the dendrite detection performance of the other three algorithms considerably more than for the training set. Figure 5 shows the motivation for decreasing this threshold. Although the diagonal of the confusion matrix presented in Table 3 indicates that only 34% of the columns were correctly identified by the nominal algorithm, the confusion matrix shows that 49% of these columns were called dendrites. Thus, the weakening of the dendrite algorithm by increasing its threshold which is accomplished (by putting a negative constant on the calculation of the detection statistic) will prevent many of these columns from being called dendrites. In this case, the column algorithm is strong enough to identify a significantly greater number of the columns correctly while the dendrite algorithm was good enough that even after decreasing its performance it still yields good performance against this data set. When the overall performance of an algorithm is unsatisfactory, examination of the confusion matrix can allow us to determine where the problem is and to suggest methods for improving the performance of the algorithm simply by adjusting the thresholds. These confusion matrices can also be used to understand where modification of the statistics associated with the projections of the classes on the Fisher direction might be adjusted to further improve their performance.

Performance on Proof Test Data Set

Figure 6 presents the complete confusion matrix for the application of the best one step algorithm to the 1500 case proof test data set. The performance of this larger set is between the training and manual referee sets.

FIGURE - 6 CONFUSION MATRIX FOR 1500 OBJECT TEST SET

CONFUSION MATRIX BASED ON COUNTS

ACTUAL CLASS	CALLED CLASS 1	2	3	4	5	6
1	214.00	8.00	37.00	92.00	2.00	39.00
2	7.00	196.00	19.00	0.0	1.00	3.00
3	120.00	13.00	142.00	13.00	1.00	54.00
4	66.00	12.00	56.00	293.00	0.0	41.00
5	0.0	0.0	0.0	0.0	1.00	0.0
6	4.00	38.00	5.00	15.00	0.0	8.00

NORMALIZED CONFUSION MATRIX

ACTUAL CLASS	CALLED CLASS 1	2	3	4	5	6
1	0.55	0.02	0.09	0.23	0.01	0.10
2	0.03	0.87	0.08	0.0	0.00	0.01
3	0.35	0.04	0.41	0.04	0.00	0.16
4	0.14	0.03	0.12	0.63	0.0	0.09
5	0.0	0.0	0.0	0.0	1.00	0.0
6	0.06	0.54	0.07	0.21	0.0	0.11

3.0 COMPARISON OF HUMAN AND MACHINE CLASSIFICATION OF POORLY DEFINED PATTERNS

The problem addressed in this report differs from many problems in pattern recognition in that a significant number of the images to be classified are so ambiguous that there is no consensus on their correct identification. Therefore, even human classification is highly dependent on the individual and the individual's physical and mental condition at the time the classification is made. It is clear that successful development of a pattern recognition algorithm which could be implemented on a computer offers many potential advantages especially with respect to consistency for a problem such as this. However, these same characteristics can also be expected to degrade the performance of the machine classifier relative to the "trainer".

The purpose of this section is to present the comparison of the compatibilities of the machine classifier which was developed for these cloud particles with the performance of human classifications.

In general, both manual and machine classification were available on the following three sets of data: 1) the 403 case training set, 2) the 201 case manual reference set, and 3) the 1500 case proof test set. Except for the 1500 case proof test set, at least two different manual classifications and two different machine classifications were made for each of these data sets. For each of these three data sets, the manual classification arrived at by the team effort and defined as the correct classification was available. For the 403 training cases, there was also the initial manual identification used to derive the first exploratory algorithms. We shall refer to this manual classification as the original training classification.

3.1 Percent Agreement Between 15-Human and 2-Machine Classifications

The performance of both the manual and machine classifiers was compared by calculating the percent agreement between each classifier and expressing this in decimal form. Table 4 presents these percent agreements for the correct and original training manual classifications and for machine classifications using the algorithms derived from the original training data and an algorithm derived from the "correct" set of training data. These two algorithms are identified as ORIG-ALG and COR-ALG, respectively.

The machine classification performances shown in Table 4 are based on a slight modification of the unbiased "one-out method" of Lackenbrach and Mickey (ref 7). The modification consists of using small groups instead of individual storms in the process.

The agreement matrix, or Table 4, not only shows the agreement of each of the classifiers with the correct classifications but also shows the agreement between each of the classifiers considered. The first two columns label the rows of the matrix. The first column provides a classifier ID number and the second column is a descriptive title for the classifier. Similarly, the first or top row also identifies the classifier using the ID number from Column 1. The first column of decimal agreements[3] under ID of 1, lists the agreement between each of the classifiers and the correct answer, the second column, identified as 2 compares the performance of the second classifier with each of the other classifiers and so forth. Similarly, each of the rows compares the performance of the classifier identified to the left of that row with each of the other classifiers. Thus, the matrix is a symmetric matrix with a unity diagonal. To simplify the reading of the table, we have only shown the lower half of the matrix since the upper half presents the same information.

Table 4 shows that the agreement between the correct and the original training data was of the same order as the agreement between the machine classifications and the correct identifications. Realizing that the original training classifications were made by experienced meteorologist in the field who was a member of the two person team to select the correct classifications, it becomes apparent that there is a major problem in obtaining consistency with manual classifications in addition to the problems of fatigue and effort required to evaluate many hundreds of thousands of images.

Table 5 presents information similar to Table 4 for the 201 case reference manual test set. For this test set, there were 14 sets of manual classifications in addition to the "correct" identification of the particles. There are also the same two machine classifications as shown in Table 4. In addition to the team effort to define the correct classifications each of the authors of this paper made their own classifications. It should be pointed out that these classifications were made by the authors after numerous discussions both of the problem of consistency in classification and general agreement among

(3) Note, decimal agreement may be interpreted as a weighted average of the diagonal of the confusion matrix, see Section 2.5 for details.

TABLE 4

AGREEMENT MATRIX COMPARING PERFORMANCE OF MANUAL AND MACHINE CLASSIFICATION OF 403 TAINING CASES

		1	2	3	4
1	CORRECT	1.00			
2	R.DYER	0.62	1.00		
3	ORIG-ALG	0.59	0.68	1.00	
4	COR.-ALG	0.73	0.55	0.60	1.00

AVERAGE ERROR IS 0.624483E 00

TABLE 5

AGREEMENT MATRIX COMPARING PERFORMANCE OF MANUAL AND MACHINE CLASS. OF 201 REF MANUAL PARTICLES

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	CORRECT	1.00																
2	R.DYER	0.67	1.00															
3	M.GLASS	0.73	0.60	1.00														
4	H.HUNTER	0.66	0.67	0.57	1.00													
5	H.H.-2ND	0.74	0.69	0.66	0.84	1.00												
6	MET-A	0.44	0.47	0.52	0.41	0.44	1.00											
7	MET-B	0.46	0.48	0.48	0.43	0.50	0.53	1.00										
8	METB-2ND	0.61	0.62	0.52	0.69	0.69	0.48	0.50	1.00									
9	MET:G-1	0.34	0.42	0.37	0.34	0.34	0.52	0.48	0.40	1.00								
10	MET:D-1	0.66	0.61	0.61	0.68	0.75	0.41	0.46	0.59	0.29	1.00							
11	TECH:D-2	0.53	0.49	0.48	0.58	0.60	0.38	0.44	0.55	0.33	0.58	1.00						
12	TECH:G-2	0.66	0.68	0.56	0.72	0.75	0.39	0.40	0.66	0.27	0.65	0.53	1.00					
13	TECH:G-3	0.65	0.65	0.67	0.68	0.73	0.49	0.49	0.55	0.43	0.66	0.52	0.66	1.00				
14	TECH:H-1	0.65	0.66	0.60	0.81	0.77	0.42	0.42	0.66	0.35	0.64	0.52	0.67	0.65	1.00			
15	H-1:2ND	0.66	0.61	0.60	0.69	0.76	0.44	0.47	0.64	0.36	0.63	0.55	0.66	0.63	0.71	1.00		
16	ORIG-ALG	0.55	0.44	0.38	0.49	0.42	0.24	0.24	0.40	0.21	0.41	0.33	0.48	0.37	0.48	0.42	1.00	
17	COR-ALG	0.53	0.39	0.36	0.38	0.40	0.20	0.25	0.34	0.14	0.44	0.32	0.40	0.33	0.37	0.41	0.5[illegible]	1.00

AVERAGE ERROR IS 0.514806E 00

TABLE 6

AGREEMENT MATRIX SHOW[illegible]G MACHINE PERFORMANCE ON 1500 PROOF TEST PARTICLES

		1	2	3
1	CORRECT	1.00		
2	ORIG-ALG	0.45	1.00	
3	COR-ALG	0.58	0.54	1.00

AVERAGE ERROR IS 0.518888E 00

each other as to how questionable cases should be handled. Despite this collusion prior to making the classifications, the agreement between the original classifications of the authors and the correct values are still less than 75%. The agreement amongst the authors is even less.

In addition to the independent classifications by the three authors, four meteorologists and four technicians were asked to manually classify the particles. The letter used to identify the technician is the first letter of the author who instructed the technician. For example, technician H-1 was instructed by Hunter. Two of the meteorologists had as part of their tasks, the task of operationally performing these classifications as the data was being gathered for notation on the data logs. It was found that, in general, there was considerably more scatter among the meteorologists then the technicians and the technicians on the average tended to do somewhat "better" than the meteorologists. However, neither group performed significantly better than the machine classifications.

It is suggested that the additional knowledge that the meteorologists have on the subject is actually a disadvantage in tasks such as this where the object is to identify a shape regardless of its meteorological implications. It is believed that some of the meteorologist who were aware of the conditions under which the data was taken where allowing their expectations to bias their decisions at least on the questionable cases. It should also be noted that in addition to the lack of agreement between the meteorologists and the correct answer, similar lack of agreement existed between all of the technicians and authors involved in this manual study. We must, therefore, conclude that, in general, it is unlikely that one would find significantly greater than 50% agreement between any one who was trained to do these classifications and the instructor. Thus, it would not be possible to solve the problem of manually classifying large numbers of particles by recruiting a large force of technicians to accomplish this unless agreements of less than 50% were acceptable.

A second question is that of how consistent is one observer with himself. To address this question, three of the observers were asked to repeat their classification of this 201 reference manual particle set at a later date. The results of the second evaluation shows agreements ranging between 50% and 84%. However, it should be noted that the two highest agreements were the author and the technician concerned with the development of the algorithms who had continued to work with the data on a daily basis and who are acutely aware of the consistency problem. Thus, we conclude that even an individual working with the data on a daily basis is highly likely to have significant disagreement between his own classifications of particles on a daily basis.

Finally, we note that on a manual reference set even the machine classifications are somewhat inferior to those which were obtained on the 403 case training set. In making comparisons across the two training sets, it is important to recognize the fact that the 403 training set was a very special set of data in which the number of particles were relatively well balanced between particle types and more importantly, the data had been selected since most were "good examples" of the particle types being examined. Thus, although they were based on real data, the 403 case set had been selected from a time period during which exceptionally well defined particles occurred. Thus, the results of Table 5 are probably more realistic.

To further investigate this question, an additional set of 1500 cases were identified as part of the effort of defining the correct variables and, in fact, both the 403 case training set and 201 case reference manual set were embedded within a total set of 2,104 cases so that the team identifying correct particles would not know at what time they were working with which group of particles. Thus, the remaining 1500 cases provided an additional set of data for which correct answers were known and which could easily be processed through the machine classifiers to provide information regarding the homogenuity of the 201 reference manual particle set. Table 6 compares the performance of the two machine classifications and the correct manual identification for the 1500 additional proof test cases. Here we see a slight improvement of the machine classifications relative to Table 5 but still significantly less than that observed in Table 4. From the examination of these tables, we conclude:

1) that machine classification using these relatively simple algorithms can be expected to yield performances approximately equal to that which can be achieved by training either meteorologists or technicians to perform this task.

2) That machine classification has significant advantages over manual classifications in terms of self-consistency in addition to the obvious advantage in fatigue and cost.

3) If one uses the machine identifications as "correct" rather than the results of a particular manual classification self-consistency implies 100% "correct" classification.

3.2 Analysis of Manual Performance

The agreement of the manual classifications was examined for each of the particles. As a result of this examination the particles for which 100% agreement were identified and those particles for which the least agreement was achieved were also identified. Figure 7 presents typical examples of those particles for which everyone or at least nearly everyone was in agreement. Figure 8 presents typical examples of those particles in each type for which the greatest disagreement occurred. In both of these, the class identified first is the correct classification of the particle and the second is the classification given by the plurality of the manual classifiers. Obviously, in the case of Figure 7, these two are in agreement. For Figure 8, we have several particles for which they are in agreement and several for which they are not. If they are in agreement that indicates that the plurality of the classifiers (not including second tries by the same observer) did agree on the correct classification, however, this was never more than three observers. In the cases where they are different, the plurality was in favor of an identification which disagreed with the correct identifications.

Examination of Figure 8 shows why there is such great difficulty in getting agreement on these particles. Since we are dealing with shape, size should not influence our decision as to the particles class. However, the same shape discontinuity on a large particle may indicate the broken arm of a dendrite, whereas on a smaller particle this irregularity might be due to the finite size of the pixels. The size of particle in relation to size of discontinuity at which these two different phenomena occur is a very subjective decision and will differ from one person to another. Similarity, the length to diameter ratios at which one identifies an object as a needle, column or plate, respectively, are also quite subjective and will vary from individual to individual and may also vary according to roughness of the particle or even orientation of the particle for different human classifiers.

3.3 Analysis of Machine Performance

The performance of both the manual classifications and the two machine classifications on the 18 particles which were shown on Figures 7 and 8 shows that, in general, the machine had less difficulty with the difficult objects then the people did.

FIGURE -7 CLOUD PARTICLES WHICH PROVED EASY TO CLASSIFY MANUALLY

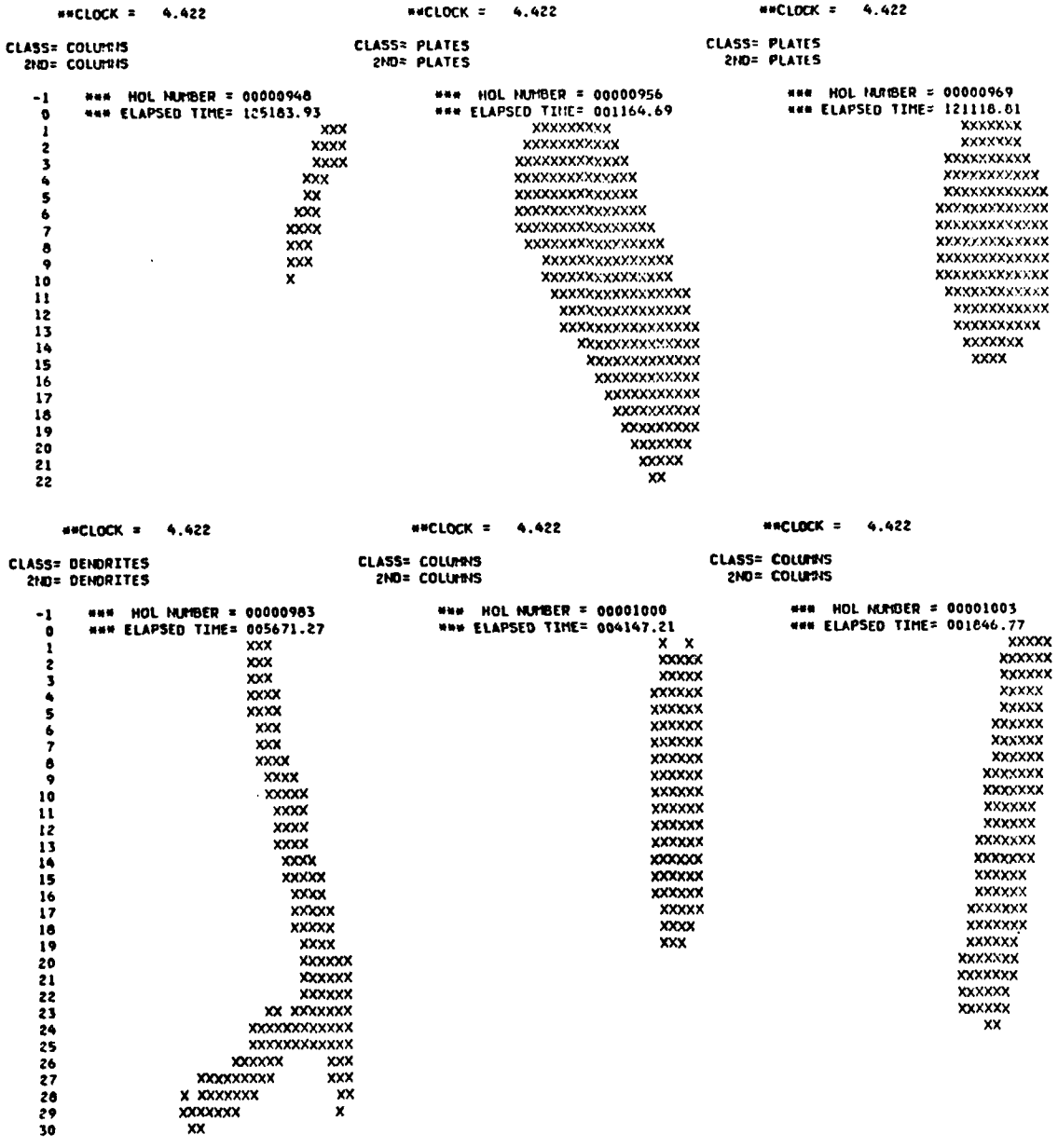


FIGURE -7 (CONT.) CLOUD PARTICLES WHICH PROVED EASY TO CLASSIFY MANUALLY

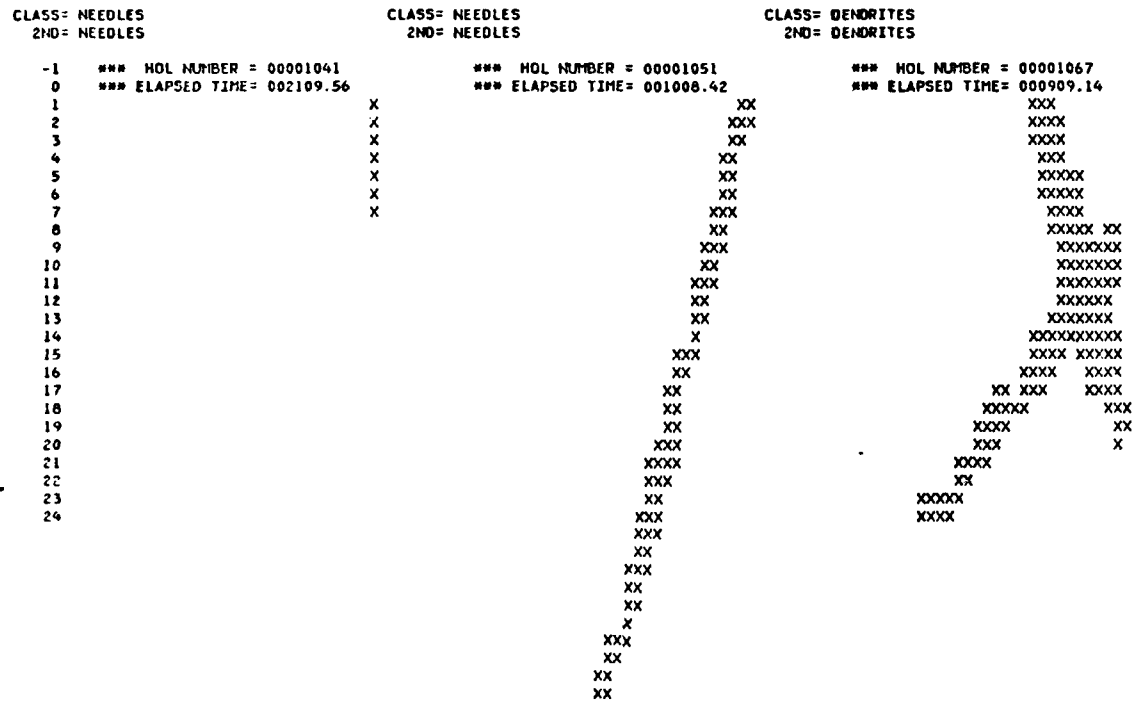
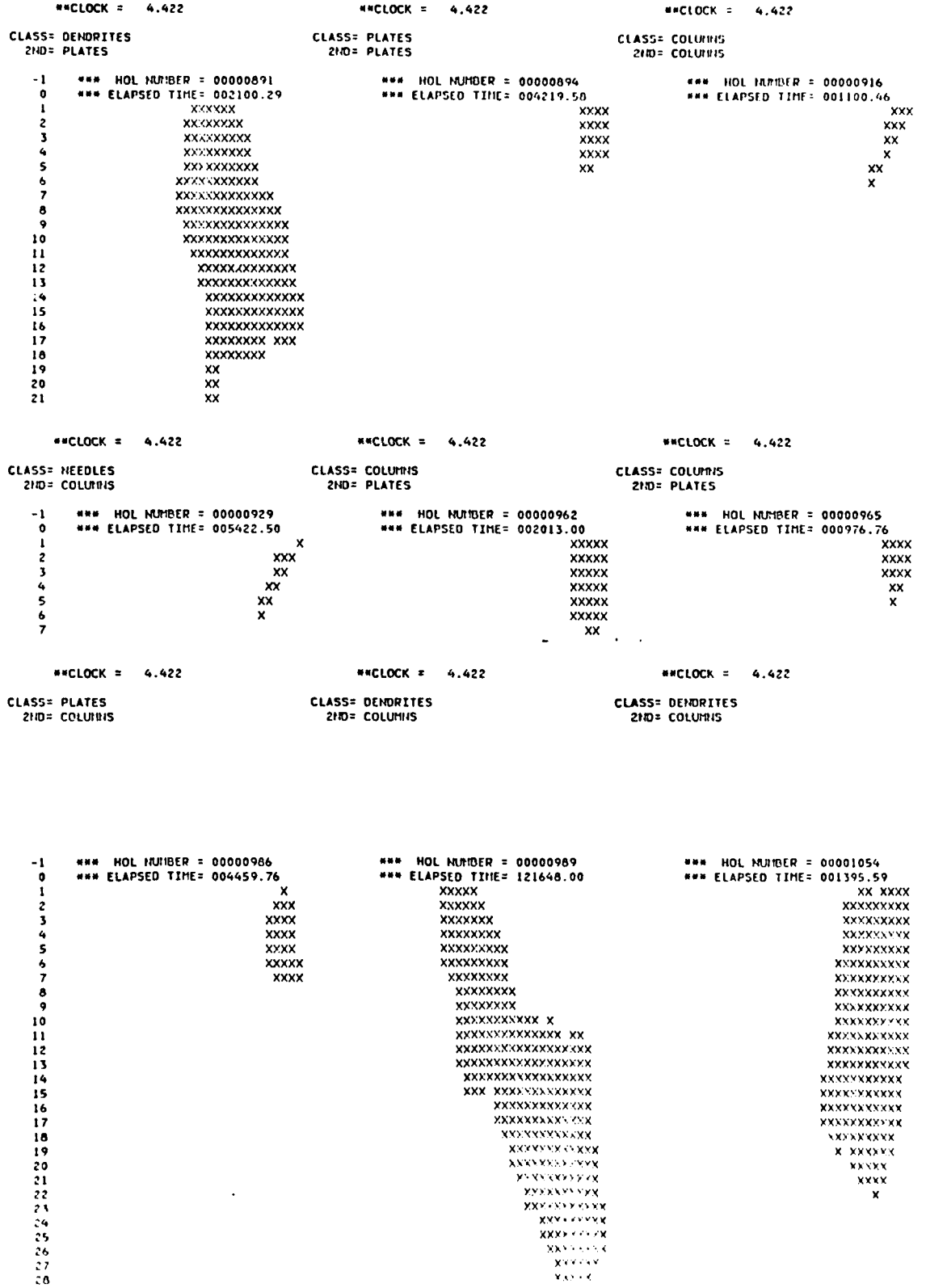


FIGURE - 8 CLOUD PARTICLES WHICH PROVED DIFFICULT TO CLASSIFY MANUALLY



Additional insight into the performance of the machine classifiers can be obtained by presenting the confusion matrix for the machine classifications. The confusion matrix (See Section 2.5) has the actual class along one axis and the class which the particle was called along the other axis. Tables 8 through 10 present the confusion matrices for the application of the two machine classifiers to the three data sets. The six classes shown are the five classes for which classification algorithms were developed plus a sixth unclassifiable class. This class is defined as any class for which the likelihood ratios associated with the other five classes were all less than unity.

Examination of Figures 4 - 7 shows that, in general, Class 6 is very poorly identified. This indicates that those particles which are difficult for humans to classify as defined by the correct classification set are not the same as those classes which are difficult for the machine to classify. It should also be noted that the classification of the columns is significantly worse for the reference manual set and the 1500 case proof test set then the 403 case training set. Since these training performance are based on the unbiased one-out method of Lackenbrach and Mickey (Ref 7) and since they occur more for the classification of columns then any other object, we conclude that this is a real effect and not a difference between performance on training data and real data. In particular, we believe that this difference is due to the character of the data sets. The training set was picked as a "clean set". Thus, the particles were more clearly defined in particular there are significantly less marginal cases between plates and columns and between columns and needles. There were also less marginal cases between plates and dendrites and columns and dendrites. However, since the columns were effected by three of the uncertainties, the effect of cleaning up the data set can be expected to be greater on columns then any of the other particles.

To illustrate the types of particles for which the machine is having the greatest difficulty, the particles having the lowest likelihood ratio are shown in Figure 9. This figure shows the types of particles which are difficult for machine classification.

FIGURE - 9A EXAMPLES OF UNCLASSIFIABLE PARTICLES SELECTED AS UNCLASSIFIABLE BY THE MACHINE CLASSIFIER

FIGURE - 9B EXAMPLES OF COLUMNS SELECTED AS UNCLASSIFIABLE BY THE MACHINE CLASSIFIER

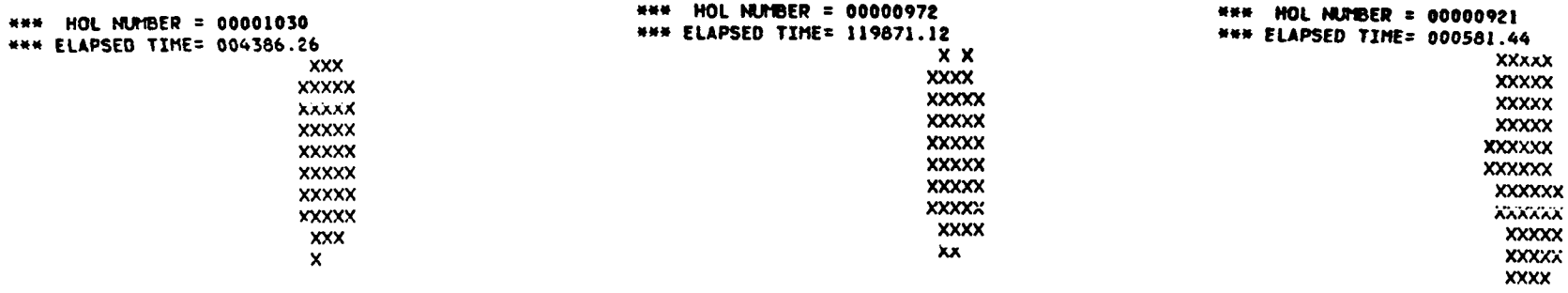
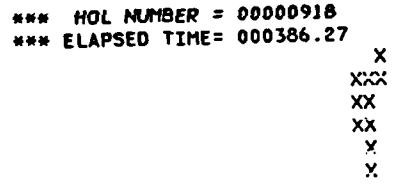


FIGURE - 9C EXAMPLE OF NEEDLE SELECTED AS UNCLASSIFIABLE BY THE MACHINE CLASSIFIER



4.0 USE OF COMPUTER PROGRAMS

Two computer programs were prepared to implement the algorithms developed in this study on the CDC computer at AFGL. These programs were prepared and debugged on the IBM computer normally used by the ADAPT Service Corporation and then delivered at AFGL. AFGL programmers then modified these programs to run on the AFGL CDC computer. The two programs prepared were a data preparation program and the classification program. Source decks and listing have been delivered to AFGL under separate cover.

4.1 Data Preparation Program

The data preparation program was a relatively minor modification of the existing AFGL data preparation program designated as KN2UTIL. The modifications of this program consisted of modifying the main and adding a new sub routine, CLANT to perform the preparation of the vectors into the standard ADAPT "ANT" format suitable for the ADAPT processing. This program also was used to perform the preliminary pre-processing which was defined in Section 2.1 of this report. This approach was taken to minimize the modifications that will be required to implement this on the AFGL CDC computer.

These modifications resulted in the addition of the following input variables to the KN2UTIL program: 1) ANT, 2) LTP, 3) LMAX, 4) TMAX, 5) HOLS, 6) IZ, 7) NGAP, 8) MIN, 9) NER, 10) REJH, 11) REJP, 12) REJC, 13) NFIND, 14) NVREG. These new variables may be input in the same manner as the other inputs used in KN2UTIL. These variables have the following meanings:

1) ANT - This is a logical variable preset to true which when true results in the preparation of a standard ADAPT input tape containing integer representations of all particles passing the selection criteria in the standard ADAPT ANT format. This output tape is referred to as the integer ANT tape.

2) -LTP - Unit upon which the integer ANT tape is written.

3) LMAX - This variable (preset to 1024) is the maximum length of diodes which will be accepted before the particle is rejected.

4) TMAX - (Preset to 85 deg.-F) Maximum temperature allowed. Particles having greater temperatures then TMAX will be rejected.

5) HOLS (preset to 1.0) Starting HOL number or count for the ADAPT ANT type vectors.

6) IZ - Three dimensional array indicating component numbers of the ANT vector to be included in the header documentation (i.e. Z1, Z2) of the ANT vector. This parameter does not effect the data processing but merely controls what information is included in the standard ADAPT documentation format.

7) NGAP (preset to 1) Number of unobstructed diodes between two regions of the image which will cause rejection based on more than one particle in the field of view.

8) LMIN (preset to 3) Minimum length of particle which will be retained.

9) NER (preset to 10) Number of tape errors which will be accepted without aborting the run.

10) REJH - Logical variable preset to true. Option to reject particles entering horizontally.

11) REJP - Logical variable preset to true. Option to reject all precipitation probe particles.

12) REJC - Logical variable preset to false. Option to reject all cloud probe particles.

13) NFIND (preset to 999999) Maximum number of acceptable particles to be found before the run is terminated.

14) NVREJ (preset to 1) The number of vertical columns of blanks which if appear between two portions of the image will be sufficient to define the image as consisting of two separate particles and thus result in the rejection of the image.

The output of this modified KN2UTIL program includes all of the outputs of the original AFGL version of the program, plus a tape containing the integer data histories in the standard ADAPT ANT format on the unit defined by LTP. There is also a new output summarizing the results of the pre-processing. This summary provides the following information: 1) the number of data histories found (always less than or equal to NFIND), 2) a definition of the documentation variables used and their location, 3) a summary of the number of histories rejected for each of the rejection criteria specified for the run.

4.2 Classification Programs

The second program which was prepared to implement the algorithms developed in this study is program CLDCLASS This program takes the output tape of the revised KN2UTIL, prepares the data vectors for application of the algorithms by converting from the integer representation of the shadow graphs to a decimal value equal to the normalized log of this integer and takes the fast Fourier transform to construct the frequency space portion of the history. It also rearranges the history such that it begins with the normalized log of the integer representing the particle shape, followed by the frequency space representation, followed by the documentation variables which the user specifies. For the algorithms developed thus far, the documentation variables used are variables 95 through 99 and represent the magnitude of the normalized linear portion of the data vector, the magnitude of the normalized frequency portion of the data vector, the Feret length, Feret width and the point of maximum width, respectively. The program has been prepared with sufficient flexibility, that other documentation parameters such as the temperature, velocity, etc could be added to the data vector. This will of course require the development of a new eigenvector transformation and new algorithms.

After preparation of the data vector as specified by the user, this program will then apply the eigenvector transformation which is included as a data statement in subroutine EIGTR. It will then apply the algorithms which are included as data statements in subroutine CLDA41. Thus, the version of this program to include different algorithms merely requires the use of a new subroutine CLDA41 and to include a new eigenvector transformation merely requires a new subroutine EIGTR. It is anticipated that there may be several different versions of subroutines EIGTR and there are already three versions of subroutine CLDA41 (the preliminary, one step and the two step algorithms). These subroutine have inputs defining their version number. These version numbers must be input to the program corresponding to the subroutine which is used or the run will be aborted.

The input variables to this program can be divided into three major groups: control variables, FFT definition variables, and classification control variables. The control variables are:

1) ITP equal to the input tape unit preset to start with unit number 10 and continue modulo 1 up to maximum allowed number.

2) ICAS (NOTPMX) Number of cases on each input tape (preset to -1) except for the first input tape which is preset to MXCAS.

3) RWIND - Option to rewind tape drive after all cases read (preset to 2).

4) IO41 - Option to use IHF, HOLS, or HOLD as input of HOL number for IO41 equal to -1, 0 or 1, respectively (preset equal to 0).

5) IHF - Input HOLS to be copied from integer ANT tape prepared by program KN2UTIL must be input as integers and the program will continue to count from any input value until the next array member is found.

6) HOLS - Defines both input HOL if decimal ANT tape is used and output HOL if decimal ANT tape is prepared unless HOLD is specified.

7) HOLD - Same as HOLS except that HOLD specifies the first HOL and the parameter HDEL specifies the increment to be used to continue from this first HOL to all HOL numbers input.

8) NTAP (preset to 9) Output integer ANT tape.

9) IVTP (preset to 4) VAL tape with VMAX VMIN of documentation variables for use in equalizing data tapes.

10) IOPRT equals last output HOL to be printed (preset equal to 1) Allows one to print all values associated with a few HOL numbers as diagnostics.

11) NATP equals number of tapes to be processed for this job (preset to maximum allowed number of tapes).

12) OPCLS equals option to perform classification and prepare Y tape (preset equal to true), (Y tape is tape of all cases used represented in eigenvector space).

13) NATP, NYTP - Output ANT and Y tape drives, respectively, if less than 2 tape not prepared (preset to minus 1).

14) OPIG - Option to transform to eigenspace (i.e. get Y's).

15) OPROT - Option to rotate particle prior to calculating Y's.

16) OPITP - Option to prepare integer ANT tape identical to input histories unless particle is rotated in which case the integer ANT tape will correspond to the rotated particles.

17) OPA41 - Option to only call subroutines EIGTR and CLDA41 NCASE times (i.e. this option does not read or write data tapes but operates only with the Y tapes).

18) NOTMK - Option not to make integer ANT tape (i.e. will not call subroutine SBITA, if NOTMK equals true (preset equal to false).

19) NPTS - Defines number of points in output data history (not required if the next three variables are included).

20) IDOC - Starting point of documentation variables in data vector.

21) NDOC - Number of documentation variables in data vector.

22) IVSFT - Starting position of FFT variables in output data vector (two places must be saved at the end of the FFT for the square magnitudes associated with the magnitude used to normalize the spacial components and the magnitude used to normalize the frequency components).

The input variables which define the FFT are as follows:

1) ISFT and NFT specified the starting point (input index) and length of vector to be used to construct FFT (must equal the power of two).

2) ICUT, NCUT - Equal upper and lower cutoff of FFT (preset equal to 2, 0, respectively; that is, the first two points of the FFT are dropped to eliminate the DC term and no high frequency terms are dropped).

3) IVSFT - Specifies start of output of FFT in output data vector.

Note, that if the FFT parameters are changed from those specified when the algorithms were delivered new eigenvector transformations and algorithms will be required.

The following variables control the use of the classification algorithms. For definition of the values to be used, the reader is referred to Section 2 of this report in which the one and two step algorithms and the variations in

the thresholds are discussed. In the following, we shall briefly describe the variables which allow you to input these criteria to the program.

1) VEREIG - This variable determines the version of the eigenvector transformation to be used, This should always be "1" for the algorithms delivered at this point.

2) VERA41 - This determines the version of the classification algorithms to be used. This parameter should equal 10 for the preliminary one step algorithm, 12 for the one step algorithm and 20 for the two step algorithm. Note, that the corresponding subroutine CLDA41 must be incorporation in the run or the run will be aborted.

3) NEIG - Number of points in the data vectors used for deriving the eigenvector transformation must be equal to 99 for the algorithms delivered at the time of this writing.

4) ICL1 - (preset to unit 7 equal card punch) Output unit on which the first and second choice will be punched in the same order as the input cases. This output will be given in integer namelist format.

5) ICL2 - Unit on which a tape containing the detection statistics will be prepared if this value is less than two no tape will be prepared.

6) LR1 - Sets options for printing summary of output: if "1" only summary is printed, if "2" summary and table of likelihood ratios is printed and if "3" summary and tables of likelihood ratios and detection statistics will be printed.

7) PROB1 - This is vector having a number of components equal to the number of algorithms. Each component in this vector represents the constant in the calculation of the detection statistic for the corresponding classification algorithm. Thus, introduction of positive components reduces the threshold value for the corresponding algorithm. In this way, the relative strength of the algorithms may be adjusted as described in Section 2 of this report.

8) PROB2 - Only two components of this input are utilized. The first component PROB2 (1) defines the criteria on the ratio of likelihoods to define the miscellaneous or unclassifiable class. If 0 or negative, any particle for which no likelihood ratio is greater than 1 is considered unclassifiable for values greater than 0, the likelihood ratio of the largest class must exceed the second largest class by a percentage greater than

or equal to PROB2 (1) or the particle is considered unclassifiable. The second component PROB2 (2) is only used in the two step (i.e. version 20) of CLDA41. For two step PROB2 (2) defines the constant in the algorithm for calculating the detection statistic for the needles or columns verses plates and dendrites first step in the separation. It may be used as described for PROB1 to adjust the threshold of this algorithm.

The remaining parameters in namelist LISTCK are used primarily for diagnostic purposes and are not required as inputs to run the program.

In addition to the options to printout the likelihood ratios and the detection statistics, a summary of the classifications is printed out whenever subroutine CLDA41 is used. This summary is prepared for each set of IO41 particles examined and defines the number of particles identified as dendrites, needles, columns, plates, streakers, and miscellaneous. Table 7 presents a typical summary of the classes produced by CLDA41 for IO41 equal to 300.

TABLE - 7 EXAMPLE OF SUMMARY OF CLASSES FOR I041=300

```
***  CUMULATIVE SUMMARY OF CLASSES SINCE FIRST PARTICLE PROCESSED  ***

     THE NUMBER OF MEMBERS IN EACH CLASS BETWEEN HDL=     901 AND     1200 ARE:

CLASS-1 (DENDRITES ) =      120
CLASS-2 (NEEDLES   ) =       35
CLASS-3 (COLUMNS   ) =       35
CLASS-4 (PLATES    ) =       91
CLASS-5 (STREAKERS ) =        1
CLASS-6 (MISC      ) =       18
```



```
FIRST CALL OF 05MAY82 CLDA41-     10
THIS SUBROUTINE APPLIES THE 5 CLOUD PARTICLE ALGORITHMS IN Y SPACE.
THIS IS  *** VERSION 10*** WHICH WHICH INCORPORATES THE FINAL UNROTATED ALGORITHMS IN
THE USER MAY ADJUST THE THRESHOLD VALUES BY INPUTING TCON AS THE PROB1 VECTOR IN LIST
PRINT OUTPUT SET BY LP1= 1=> SUMMARY ONLY
                         2=> SUMMARY AND TABLE OF  LIKELYHOOD RATIOS
                         3=> SUMMARY AND TABLES OF LIKLIHOOD RATIOS AND DETECTION STA
```

5.0 DISCUSSION OF EIGENVECTOR REPRESENTATION OF PARTICLE HISTORIES

The eigenvector representation was developed to represent the 403 training particles that were supplied for this study. Prior to developing this representation, the training set was reduced to a set having zero means by subtracting the average of all 403 particle data vectors from each of the data vectors. Figure 10* shows this average cloud particle for all the data vectors. These zero mean vectors were then processed through the ADAPT eigenvector derivation program and the eigenvectors derived. The variation explained by each of the eigendirections is plotted in the lower curve shown in Figure 11. The ordinant on this curve is the percent of explained variation and the abcissa is the eigen-direction. The upper curve plots cumulative sum of the lower curve, thus, at eigendirection-2, the lower curve indicates that approximately 5% of the variation is explained by the second eigendirection, the upper curve shows that the first two eigendirections taken together explain slightly more than 95% of the variation in the data. This curve shows that the eigenvector representation is extremely efficient for representing the particle shape and size in-formation included in these data histories. A more detailed examination of the information in this curve is shown in the curve connecting the circled points. This curve uses the ordinant scale reduced by a factor of 10,000 and reveals that the eigenvectors take on a noise-like behavior after approximately the 26th eigendirection. The next major slope change occurs after the 19th eigendirection. This suggests that between the 19th and 26th eigendirections the variation is probably mostly associated with a limited number of cases (labeled "non-Global" adjustments on Figure 11) and is probably not a good region in which to develop useful algorithms.

Figure 12 presents plots of the first four eigenvectors, The abcissa of these plots is the same indexing variable that

---

* Because of the large number of figures, all of the figures for Section 3 will be found at the end of the Section.

appeared on the data vectors presented in Figure 2. That is, each abcissa index represent one of the directions in the original data space. The ordinant is the projection of the eigenvector on that original coordinate direction. Recalling that the particle shape history and its frequency spectrum are both normalized, the first 94 points include only shape information. The 95th and 96th points represent the square magnitudes of the particle shape and its frequency spectrum, respectively. The 97th and 98th directions are the Feret length and width, respectively, and the 99th direction or last point is the point of maximum width of the particle and, therefore, primarily a shape measurement. Thus, the plot of the first eigenvector shows that the greatest magnitude in this vector is concerned with variables 95 through 98 and, thus, this vector is primarily related to the size of the particle. The second eigenvector is primarily determined by the length. The third eigenvector contains significant contributions for both the size and shape portions of the history, however, the size portions to a large extent cancel one another out and, therefore, the third eigenvector is primarily concerned with the shape of the particle. Similarly, the fourth eigendirection is related to the shape.

Figure 13 presents the projection of the 403 training cloud particles onto the first and second eigendirections. The symbols used on this plot are shown under the title. The files referred to are the files on the original data tape and were originally identified as follows: File 1 was primarily rain, File 2 was primarily large snow, File 3 was primarily bullet rosets, File 4 was primarily needles and File 5 was primarily dendrites. Notice, that three particles which are separated from the main cluster identified by the symbols, 7, B, and 4. These are the only three streakers which were in the data set. It is clear that these streakers are easily separated in this scatter plot of the first two eigendirections.

Figure 14 represents a blow-up of the region in Figure 13 containing the majority of the data. The symbols on this plot show that class of the particle as determined by a more careful second examination of each of the individual particles which were used for this study. The numerals represent "pure cases" and the letter less certain identifications. This figure shows the region in which the needles represented by A's and 1's are almost all located in the lower center of the figure whereas the dendrites represented by 2's and B's are primarily located above an eigendirection 1 value of minus 10

and on the left side of the figure. Both of these classes are reasonably well separated even in this first scatter plot which explains 97.8% of the variation.

Figure 15 presents the direction of the 403 cloud particles on the third and fourth eigendirections using the symbols associated with the original data files. Figure 9 is a blow-up of the lower left hand corner of Figure 14 and uses the symbols identifying the final classes associated to the training data. These two eigendirections explain 1.14% of the variation in the data. There are no good separations on this scatter plot, however, the bullets or columns appear to be grouped together in a non-linear region.

Figures 16 through 28 present similar plots of the eigenvectors and the projections on these eigenvectors using the symbols to identify the particle according to its final training classification for eigendirections Number 5 through 20.

FIGURE 10

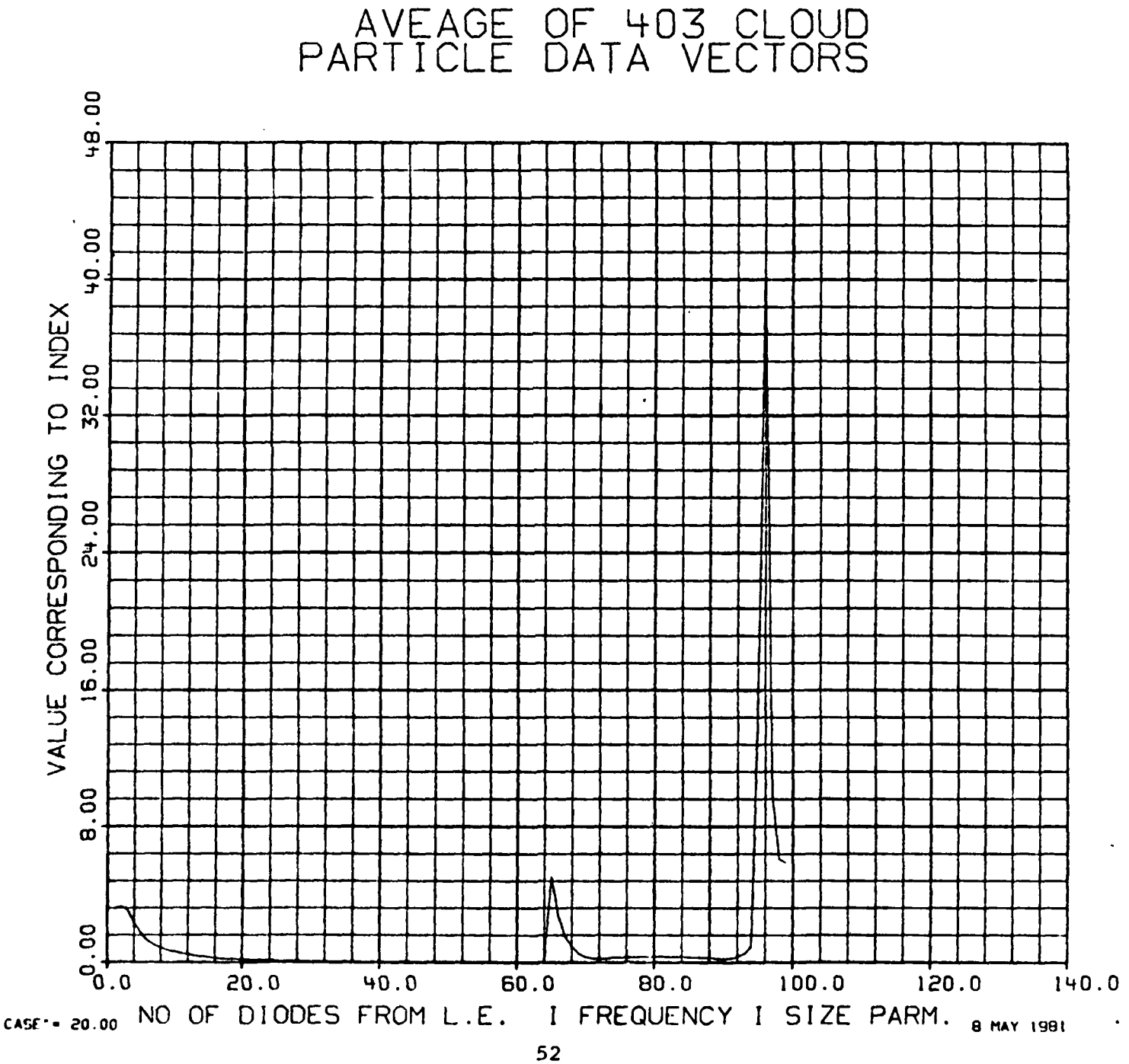

AVEAGE OF 403 CLOUD
PARTICLE DATA VECTORS
VALUE CORRESPONDING TO INDEX
0.00
8.00
16.00
24.00
32.00
40.00
48.00
0.0
20.0
40.0
60.0
80.0
100.0
120.0
140.0
NO OF DIODES FROM L.E. I FREQUENCY I SIZE PARM.
CASE - 20.00
8 MAY 1981

FIGURE 11

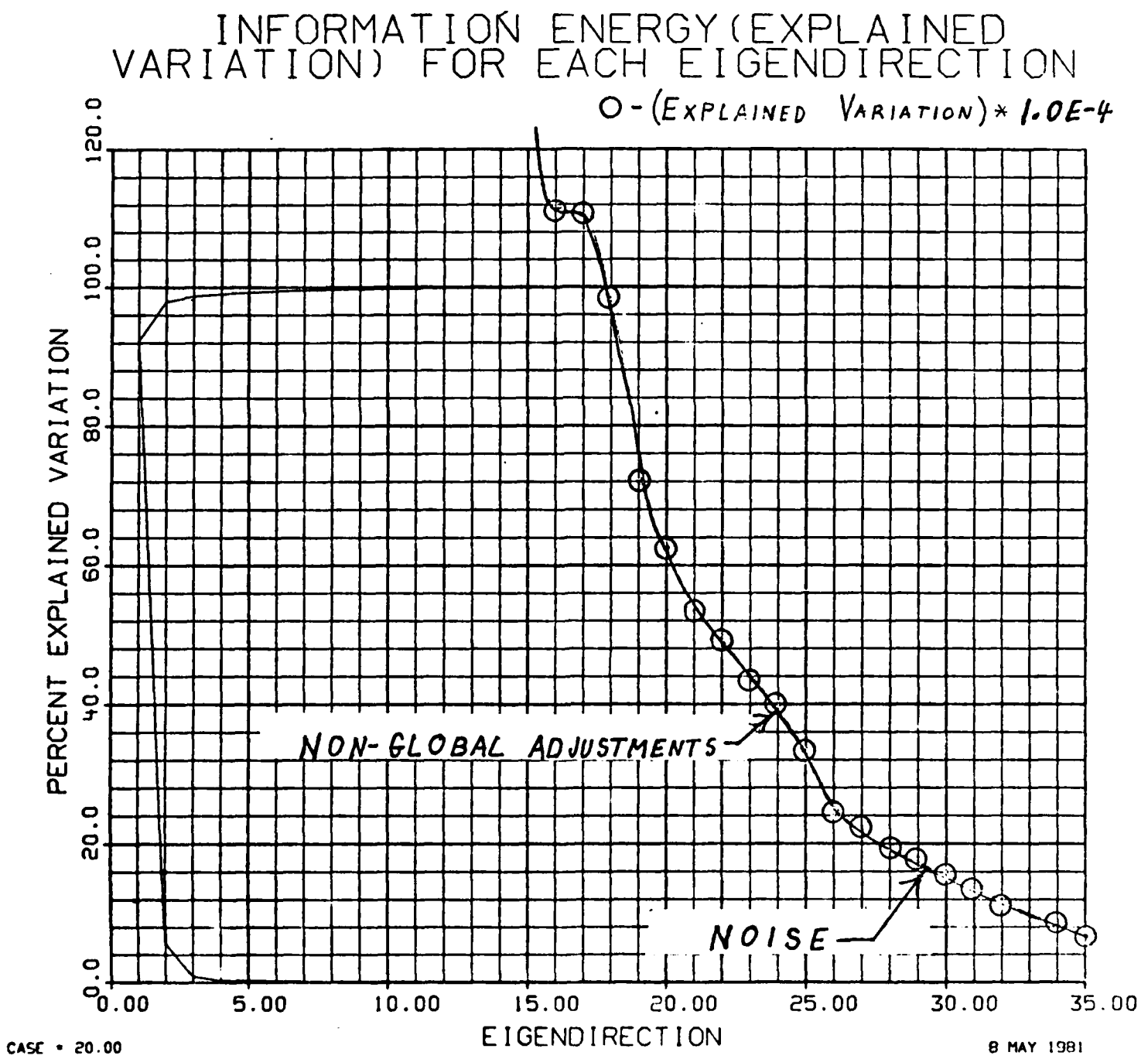
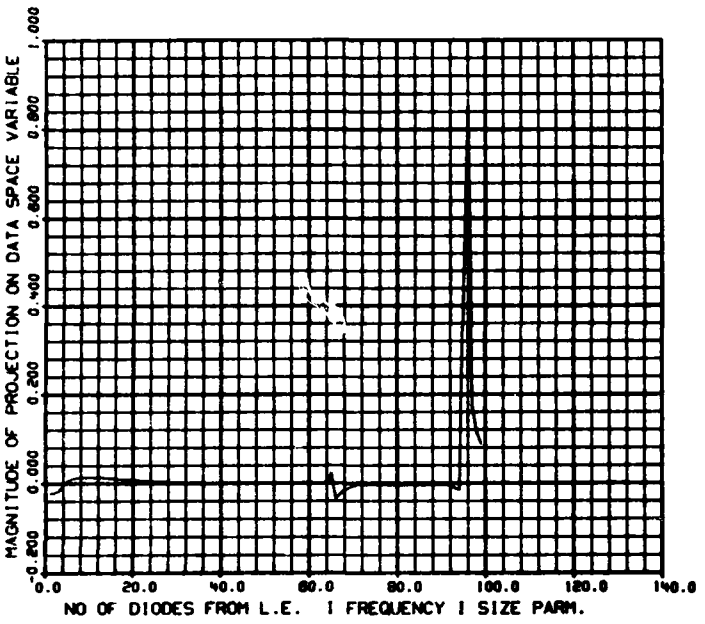
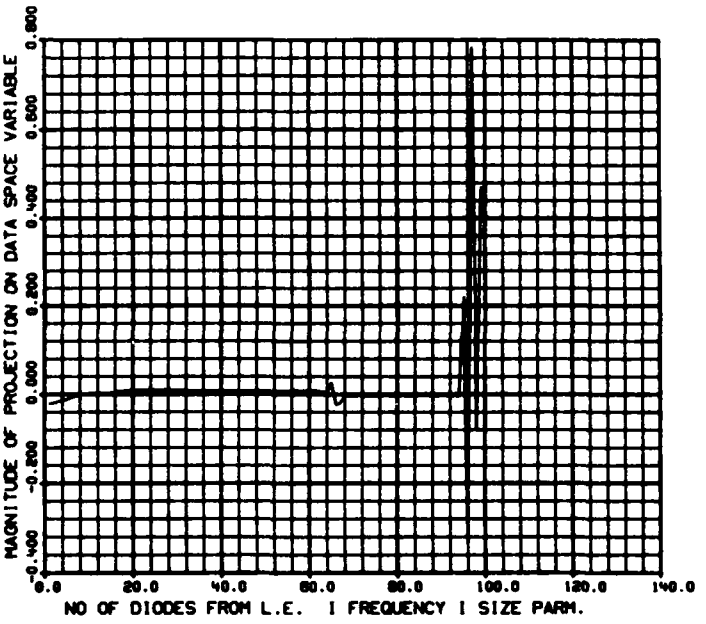

INFORMATION ENERGY (EXPLAINED VARIATION) FOR EACH EIGENDIRECTION
O - (EXPLAINED VARIATION) * 1.0E-4
PERCENT EXPLAINED VARIATION
120.0
100.0
80.0
60.0
40.0
20.0
0.0
NON-GLOBAL ADJUSTMENTS
NOISE
0.00
5.00
10.00
15.00
20.00
25.00
30.00
35.00
EIGENDIRECTION
CASE • 20.00
8 MAY 1981

FIGURE-12 EIGENVECTORS 1-4

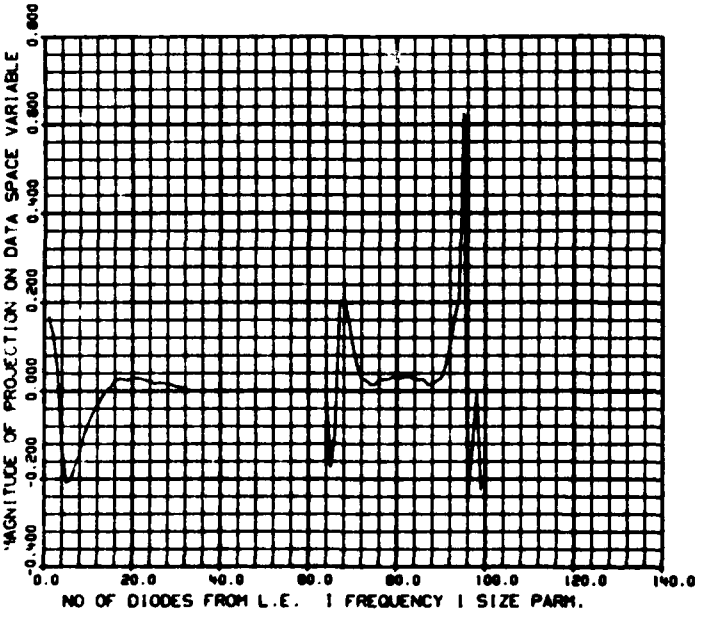
EIGENVECTOR NUMBER - 1



EIGENVECTOR NUMBER - 2



EIGENVECTOR NUMBER - 3



EIGENVECTOR NUMBER - 4

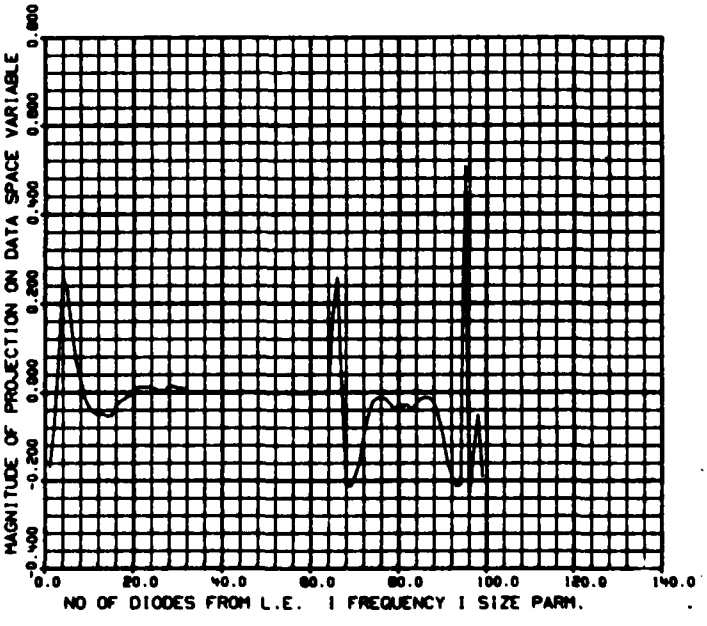


FIGURE-13

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 1 AND 2 -

(2-9,*=FILE#1; B-J=FILE#2; L-T=FILE#3; Y=FILE#4;ANDU-Z,^,<>,+,$=FILE#5)

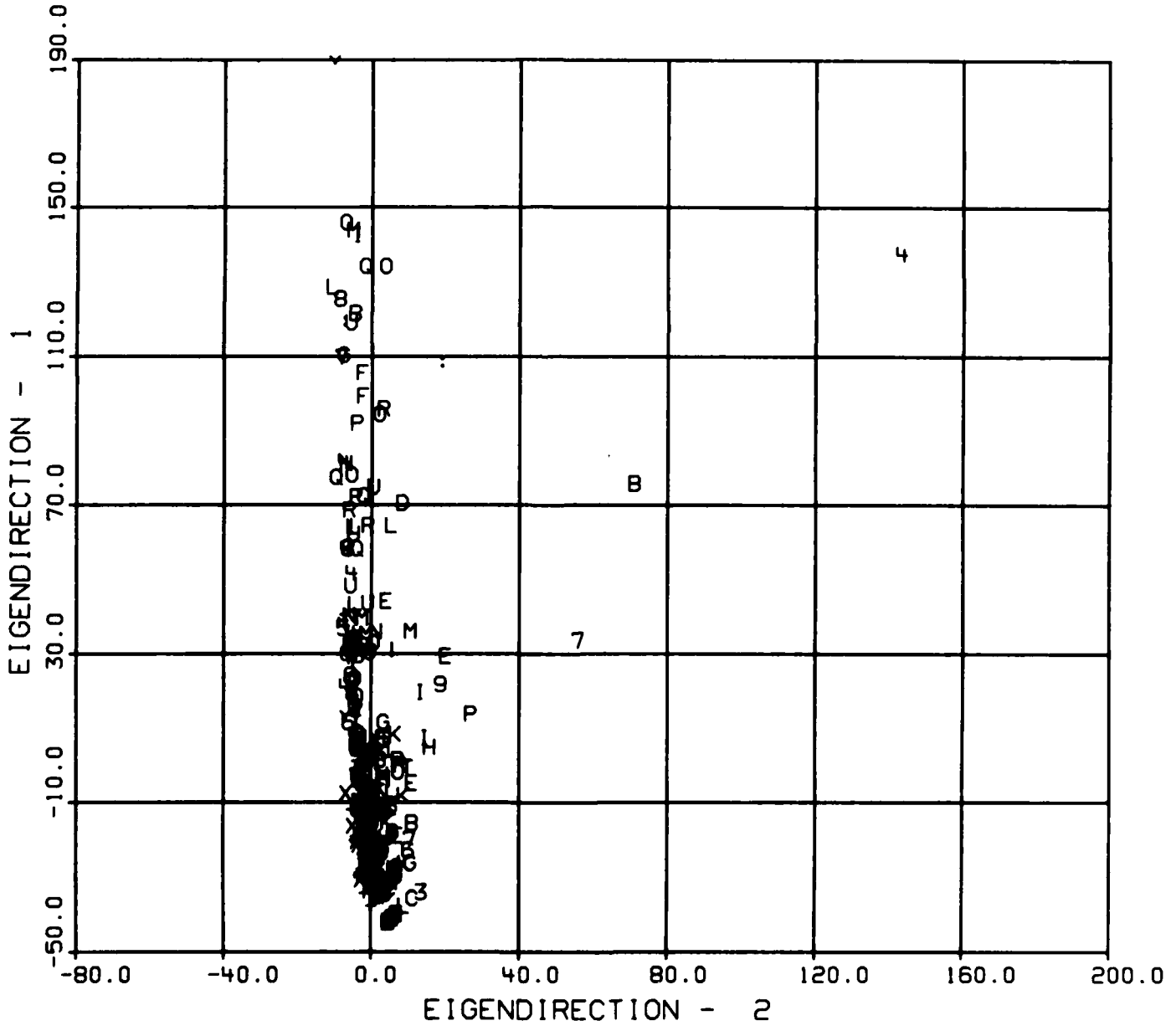


FIGURE-14

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 1 AND 2 - (1,A=NDL; 2,B=DNDRT; 3,C=BLT; 4,D=COL; 5=BLT OR COL; 6,F=PLT; 7=PLT OR SPH; 8=MISC; 9=STRKR)

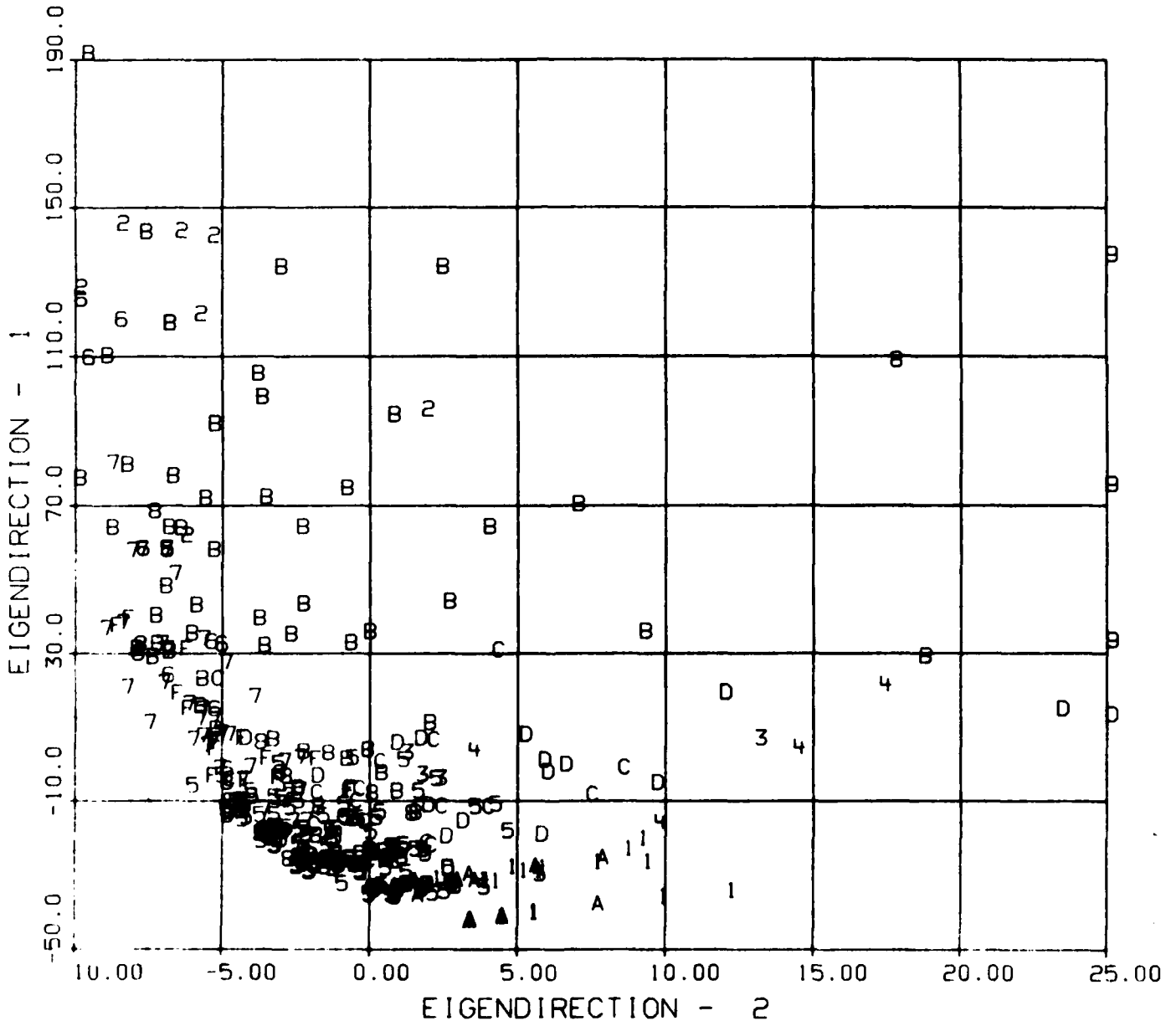


FIGURE-15

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 3 AND 4 -

(2-9,*=FILE#1; B-J=FILE#2; L-T=FILE#3; Y =FILE#4;ANDU-Z,^ ,<>,+,$=FILE#5)

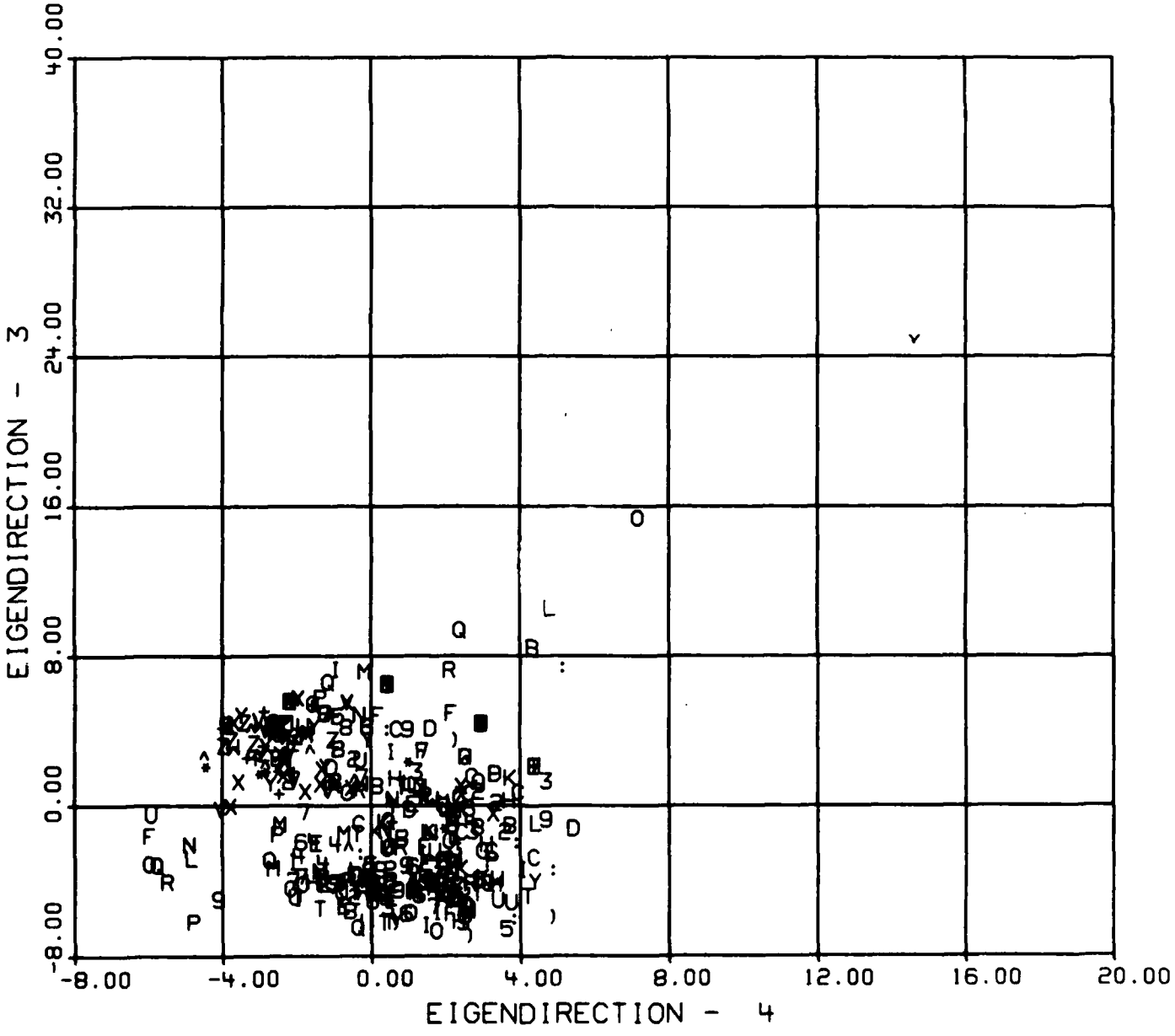
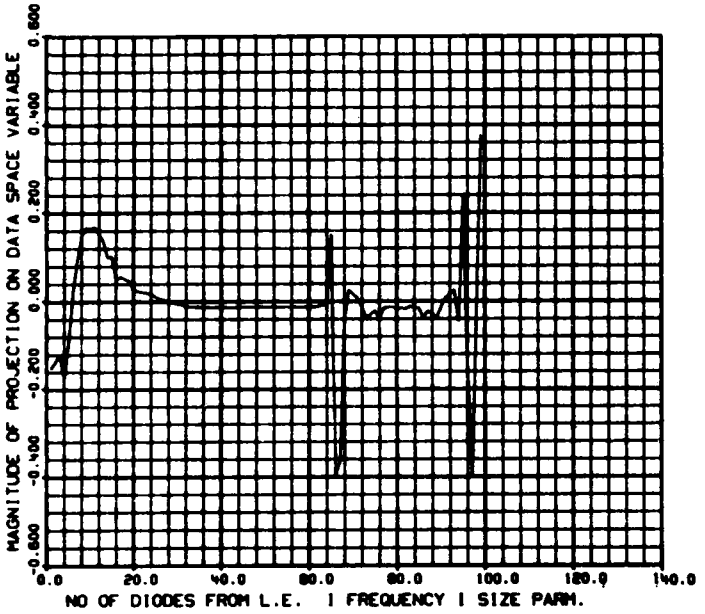
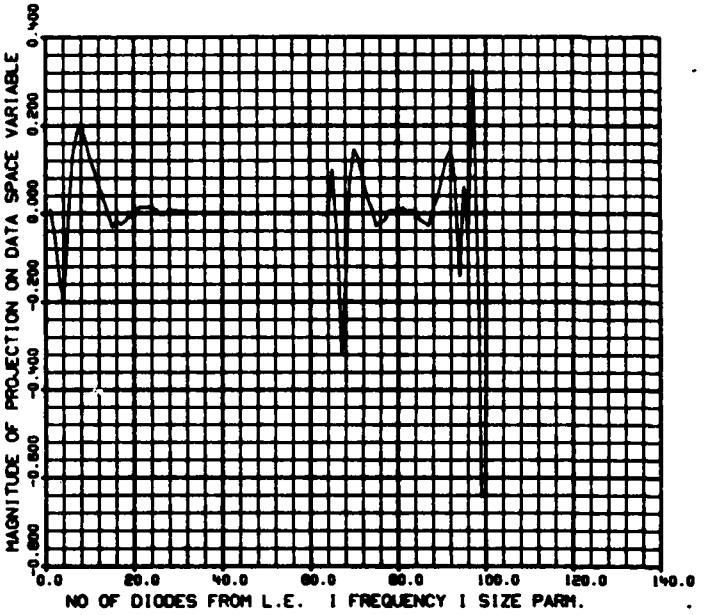


FIGURE-16 EIGENVECTORS 5-8

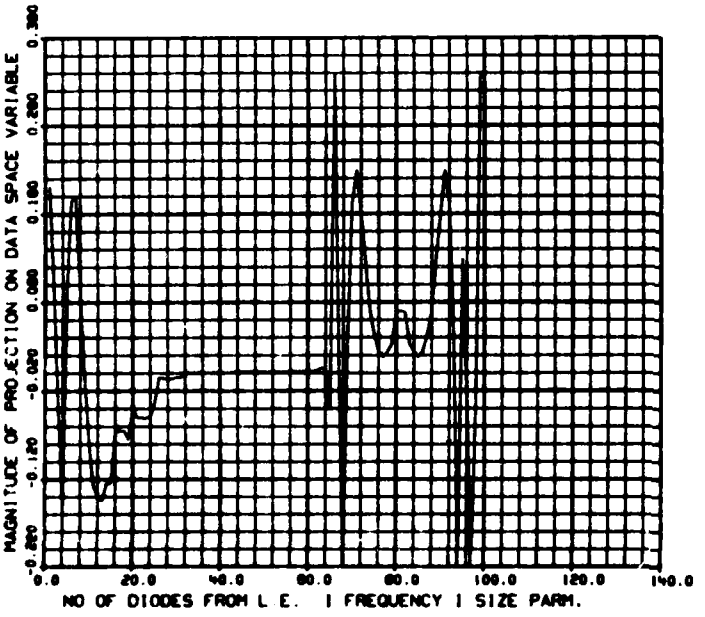
EIGENVECTOR NUMBER - 5



EIGENVECTOR NUMBER - 6



EIGENVECTOR NUMBER - 7



EIGENVECTOR NUMBER - 8

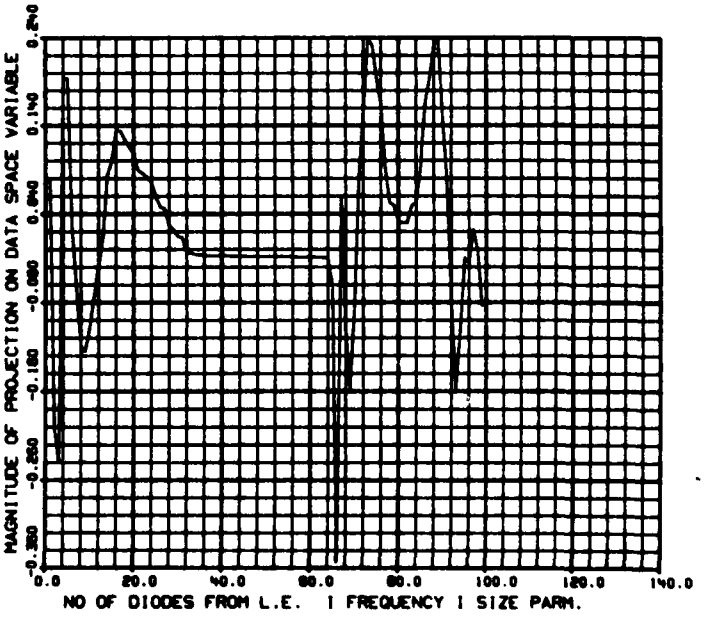


FIGURE 17

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 3 AND 4 - (1,A=NDL; 2,B=DNDRT; 3,C=BLT; 4,D=COL; 5=BLT OR COL; 6,F=PLT; 7=PLT OR SPH; 8=MISC; 9=STRKR)

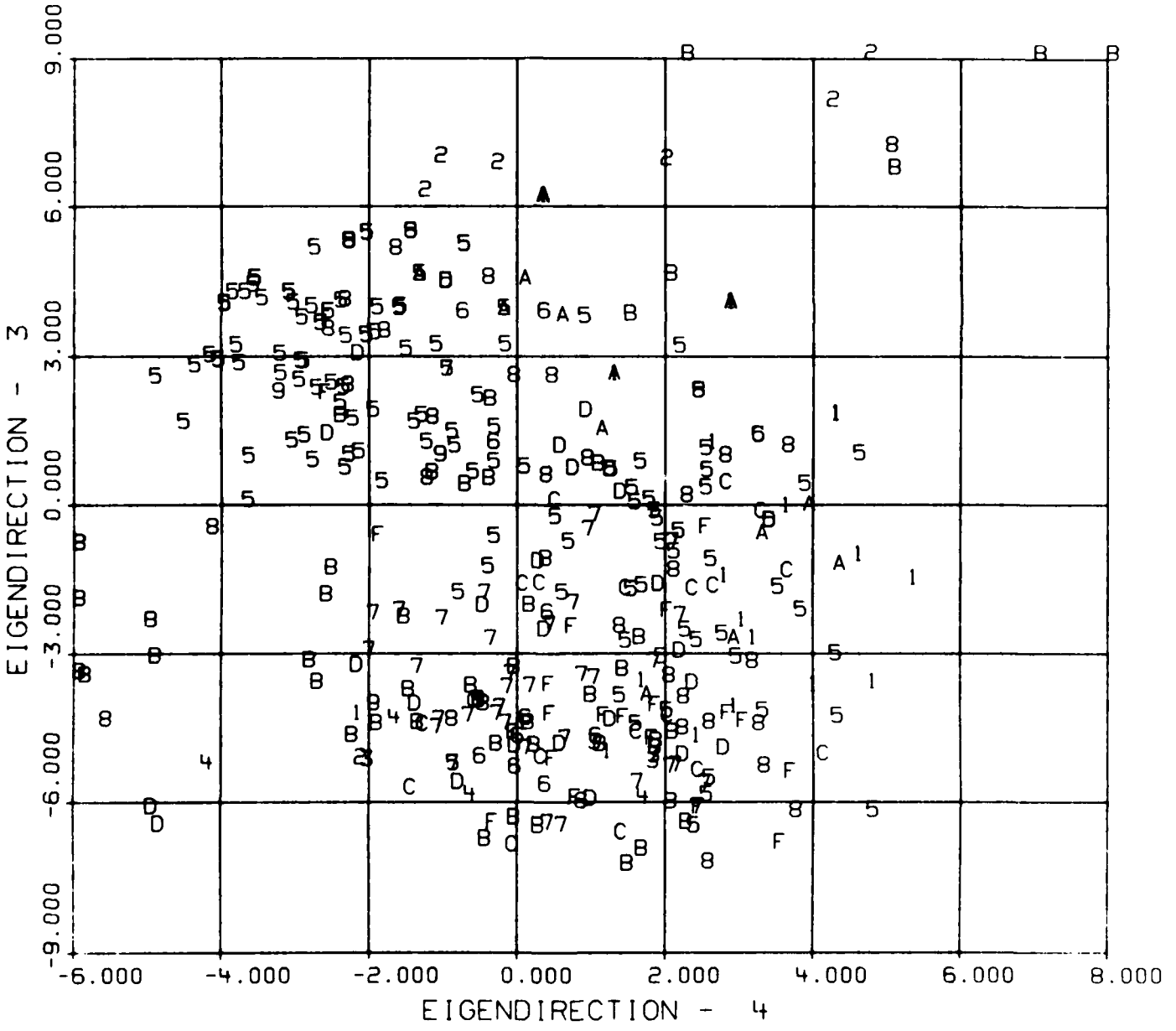
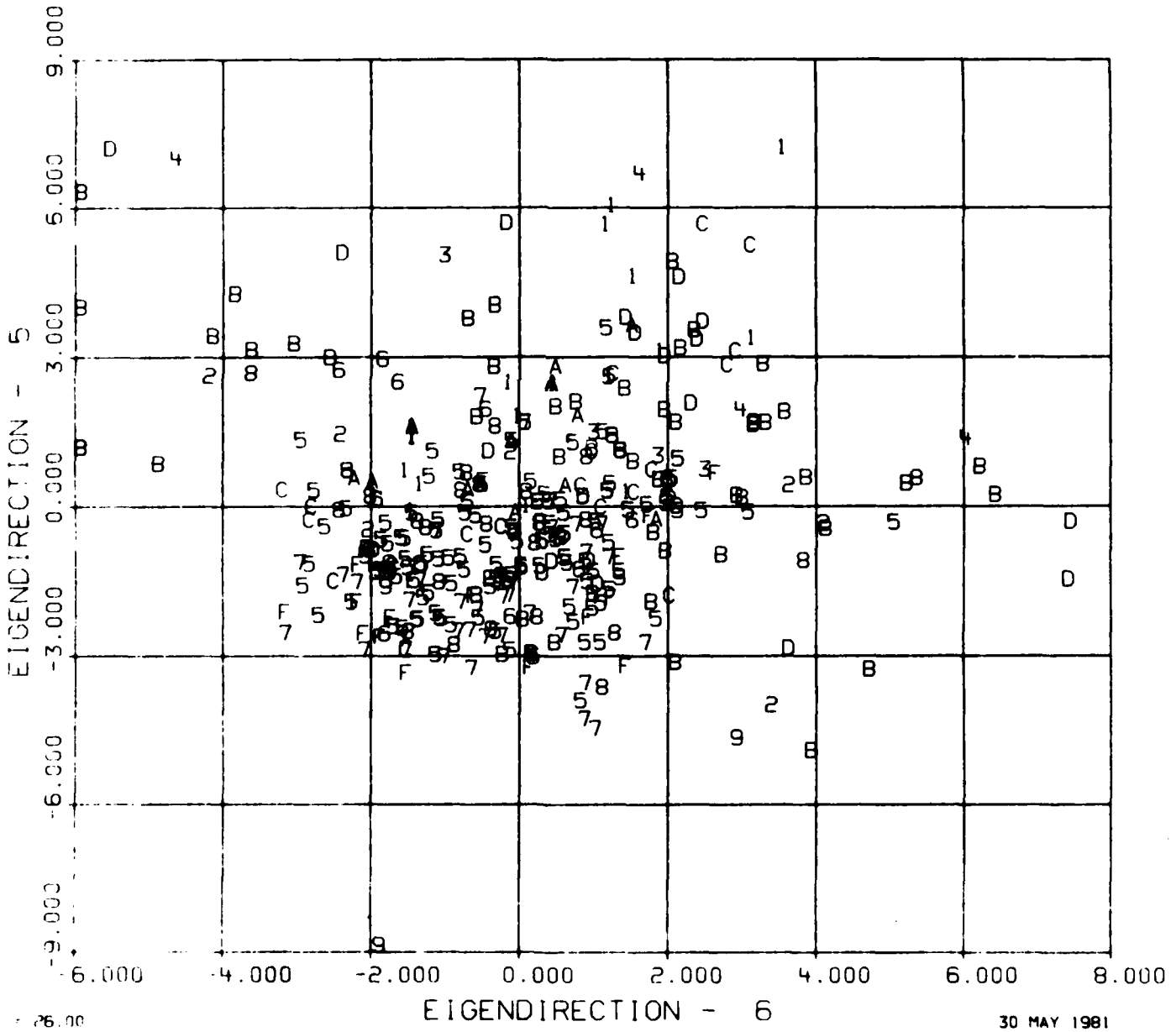


FIGURE 18

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 5 AND 6 - (1,A=NDL; 2,B=DNDRT; 3,C=BLT; 4,D=COL; 5=BLT OR COL; 6,F=PLT; 7=PLT OR SPH; 8=MISC; 9=STRKR)



CASE = 26.00

30 MAY 1981

FIGURE-19

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 7 AND 8 - (1,A=NDL; 2,B=DNDRT; 3,C=BLT; 4,D=COL; 5=BLT OR COL; 6,F=PLT; 7=PLT OR SPH; 8=MISC; 9=STRKR)

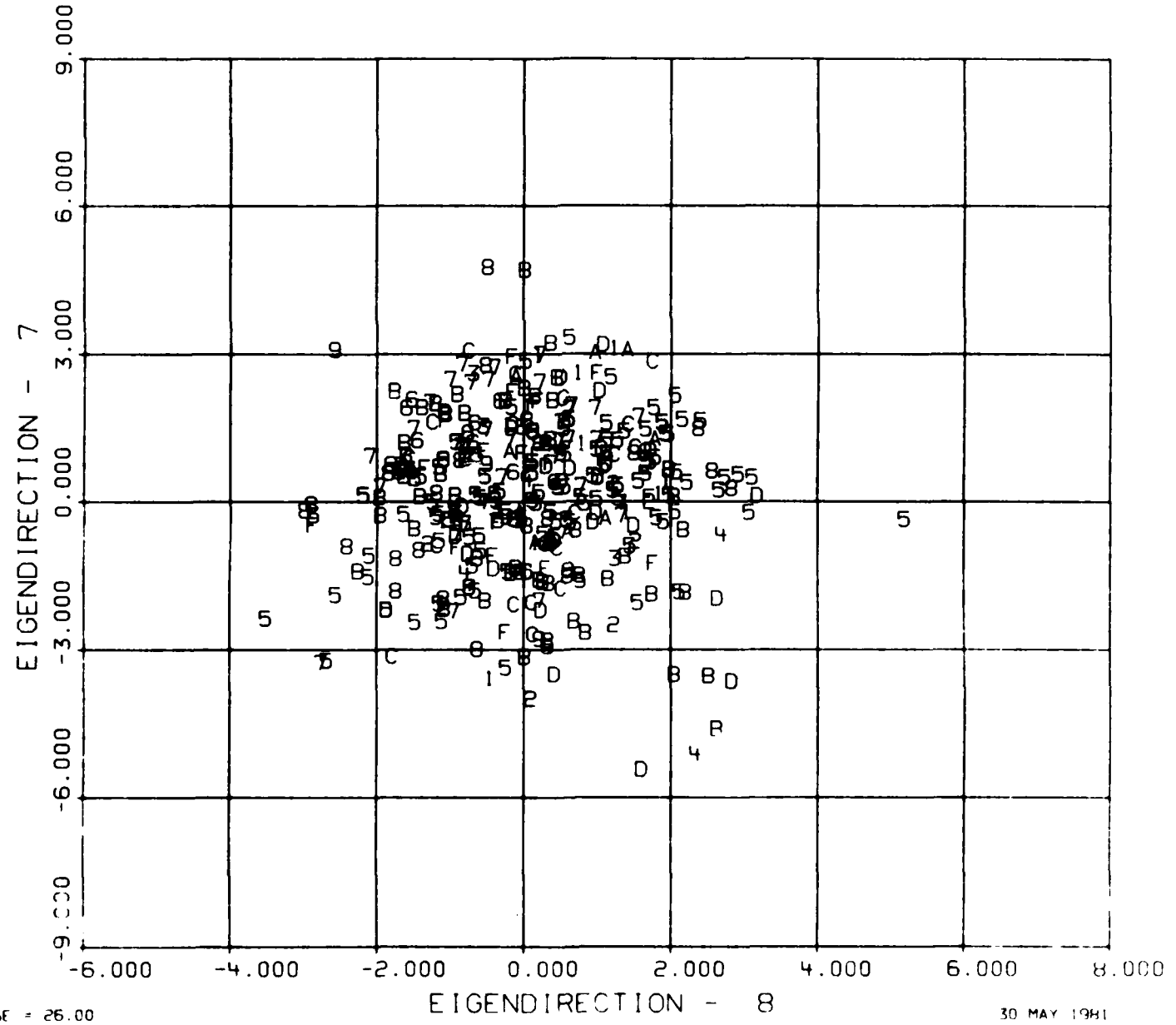


FIGURE-20 EIGENVECTORS 9-12

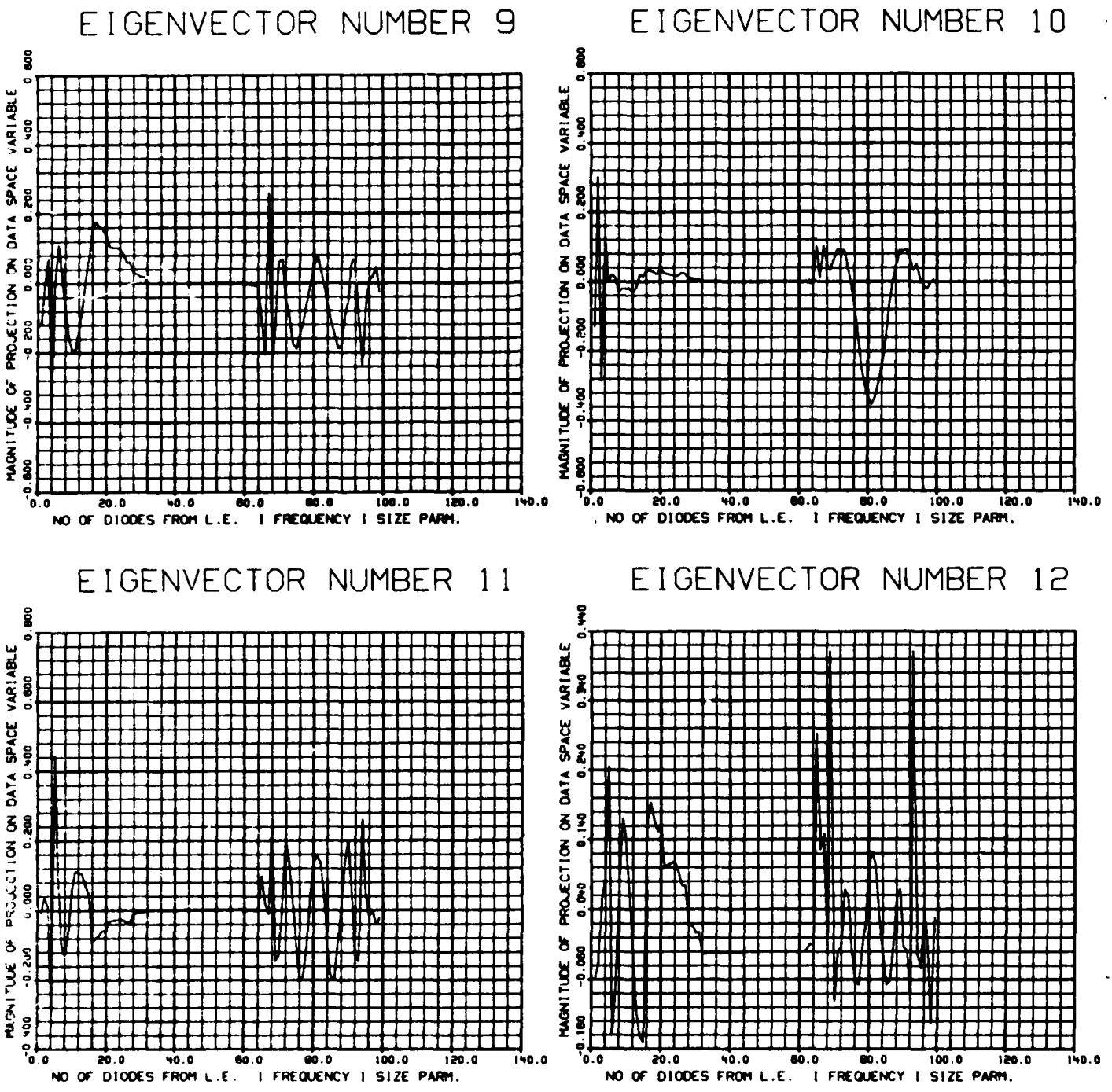


FIGURE-21

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 9 AND 10 - (1,A=NDL; 2,B=DNDRT; 3,C=BLT; 4,D=COL; 5=BLT OR COL; 6,F=PLT; 7=PLT OR SPH; 8=MISC; 9=STRKR)

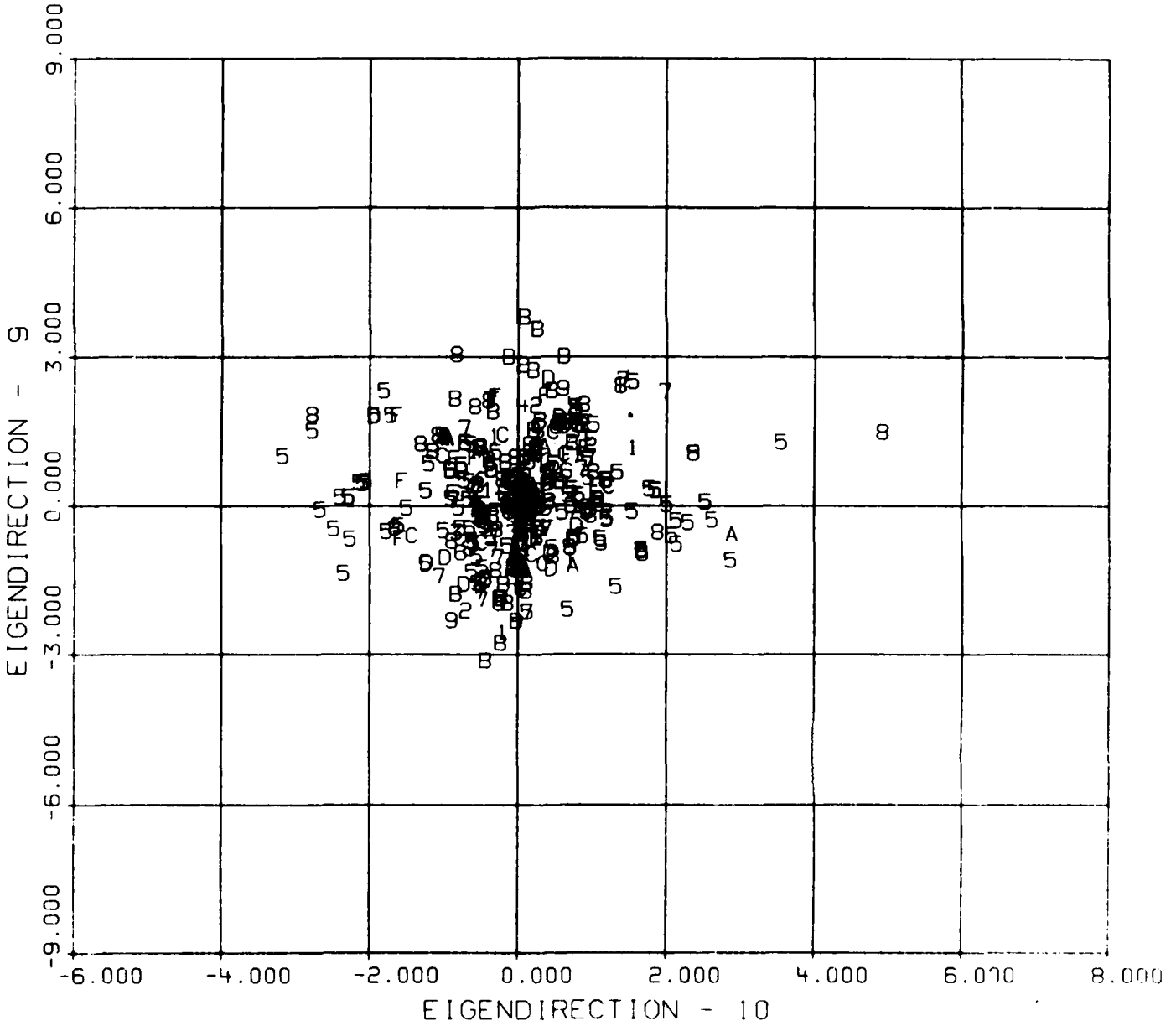


FIGURE 22

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 11 AND 12 - (1,A=NDL; 2,B=DNDRT; 3,C=BLT; 4,D=COL; 5=BLT OR COL; 6,F=PLT; 7=PLT OR SPH; 8=MISC; 9=STRKR)

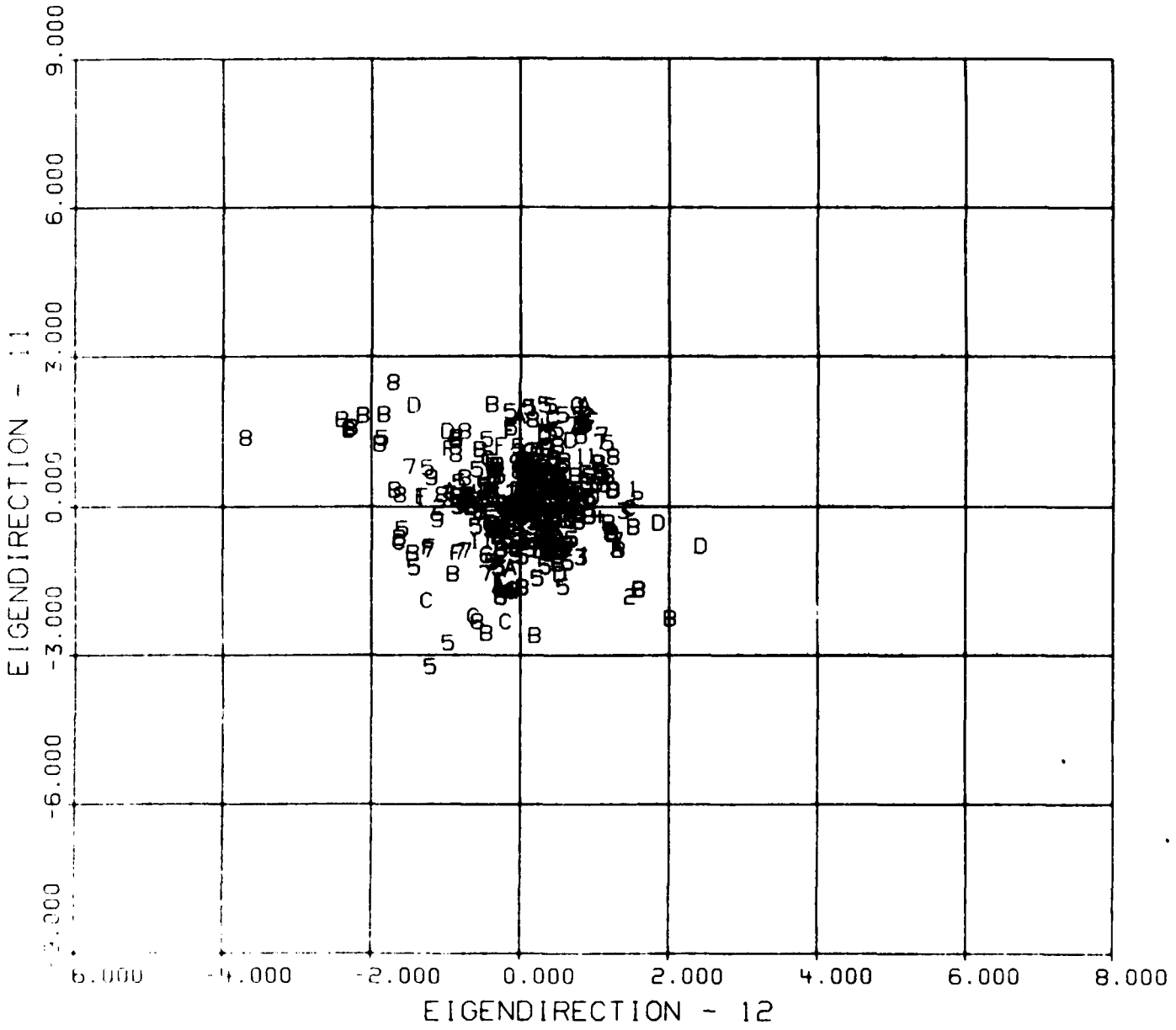


FIGURE-23 EIGENVECTORS 13-16

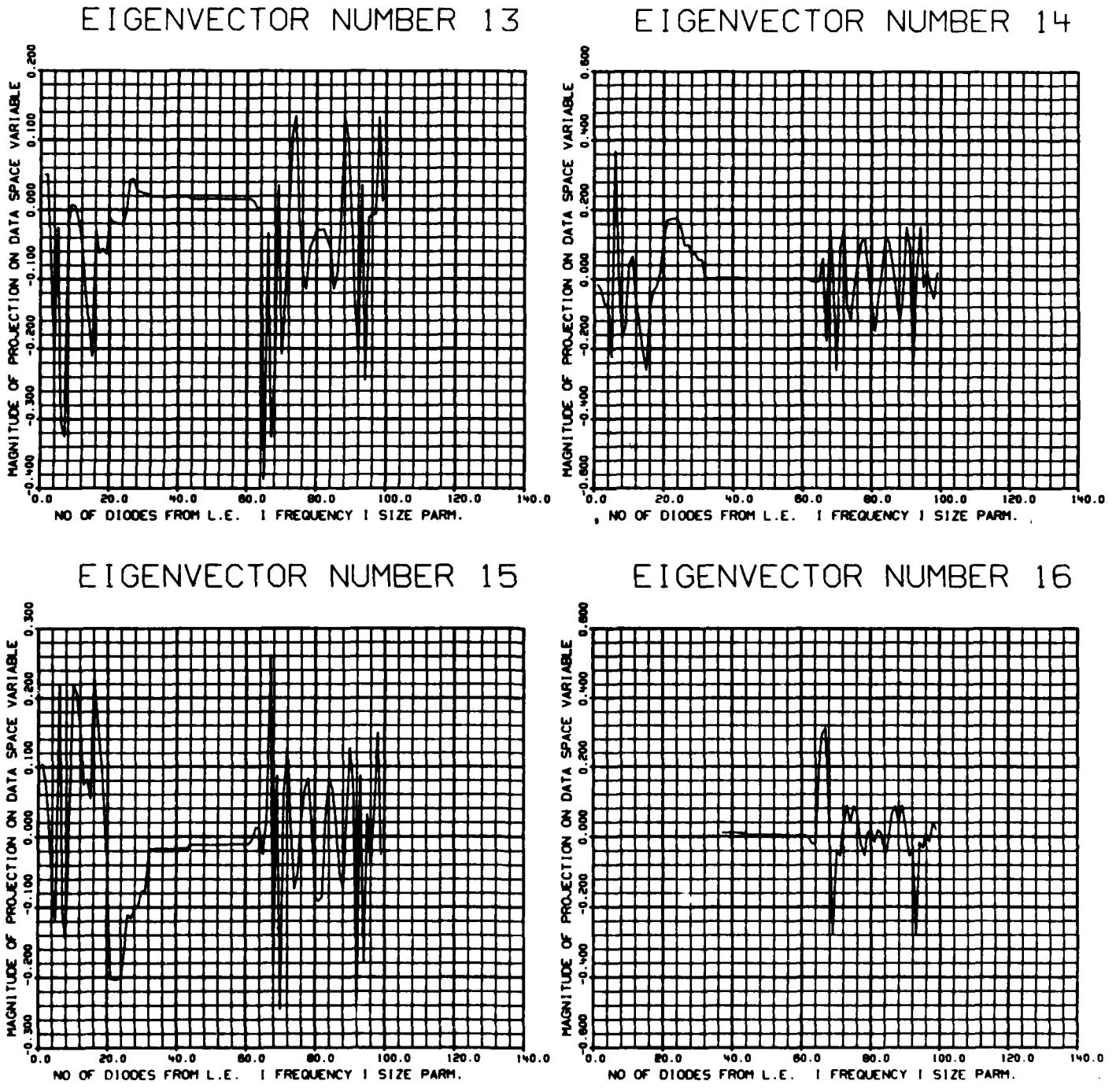


FIGURE-24

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 13 AND 14 - (1,A=NDL; 2,B=DNDRT; 3,C=BLT; 4,D=COL; 5=BLT OR COL; 6,F=PLT; 7=PLT OR SPH; 8=MISC; 9=STRKR)

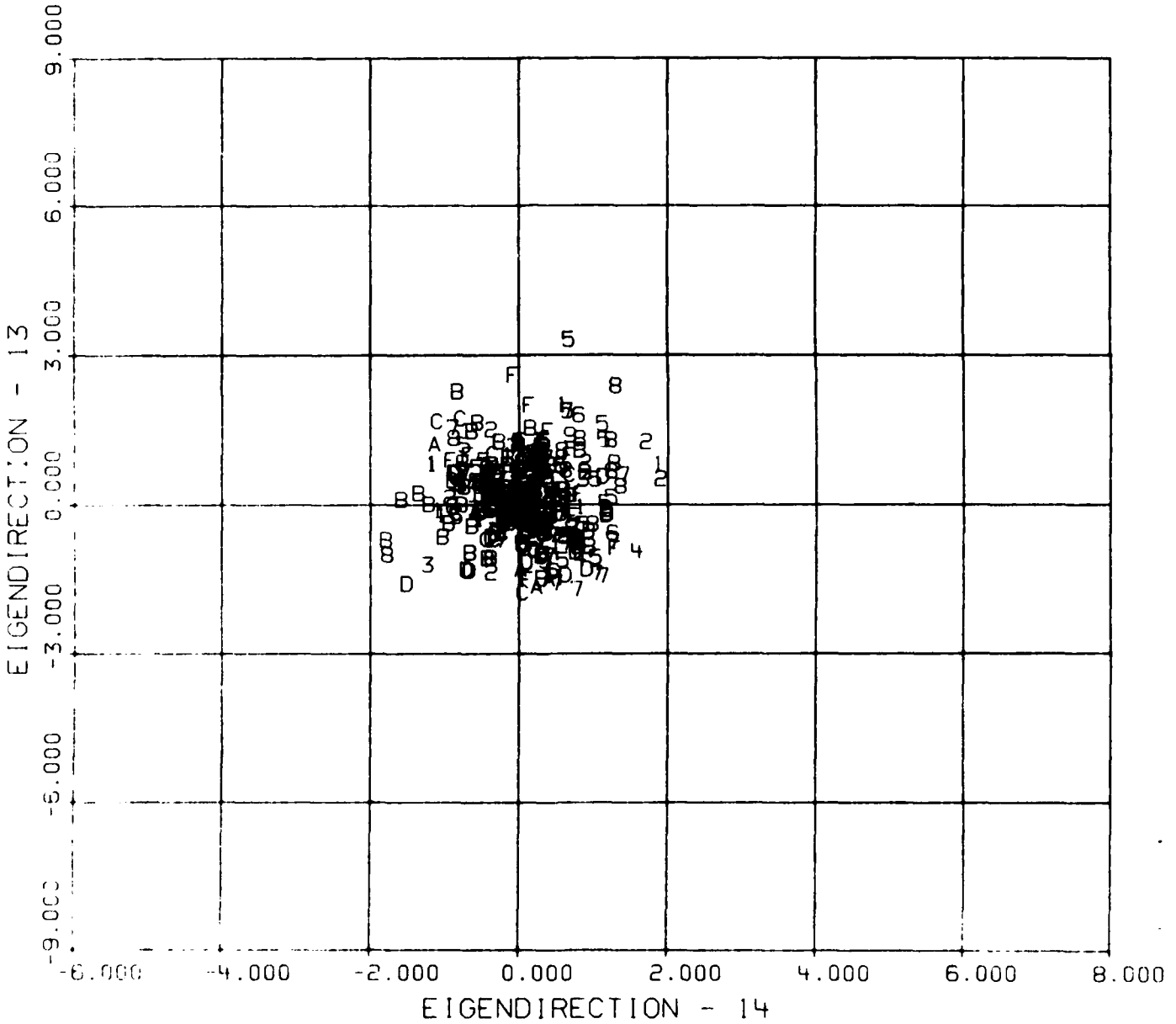


FIGURE-25

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 15 AND 16 - (1,A=NDL; 2,B=DNDRT; 3,C=BLT; 4,D=COL; 5=BLT OR COL; 6,F=PLT; 7=PLT OR SPH; 8=MISC; 9=STRKR)

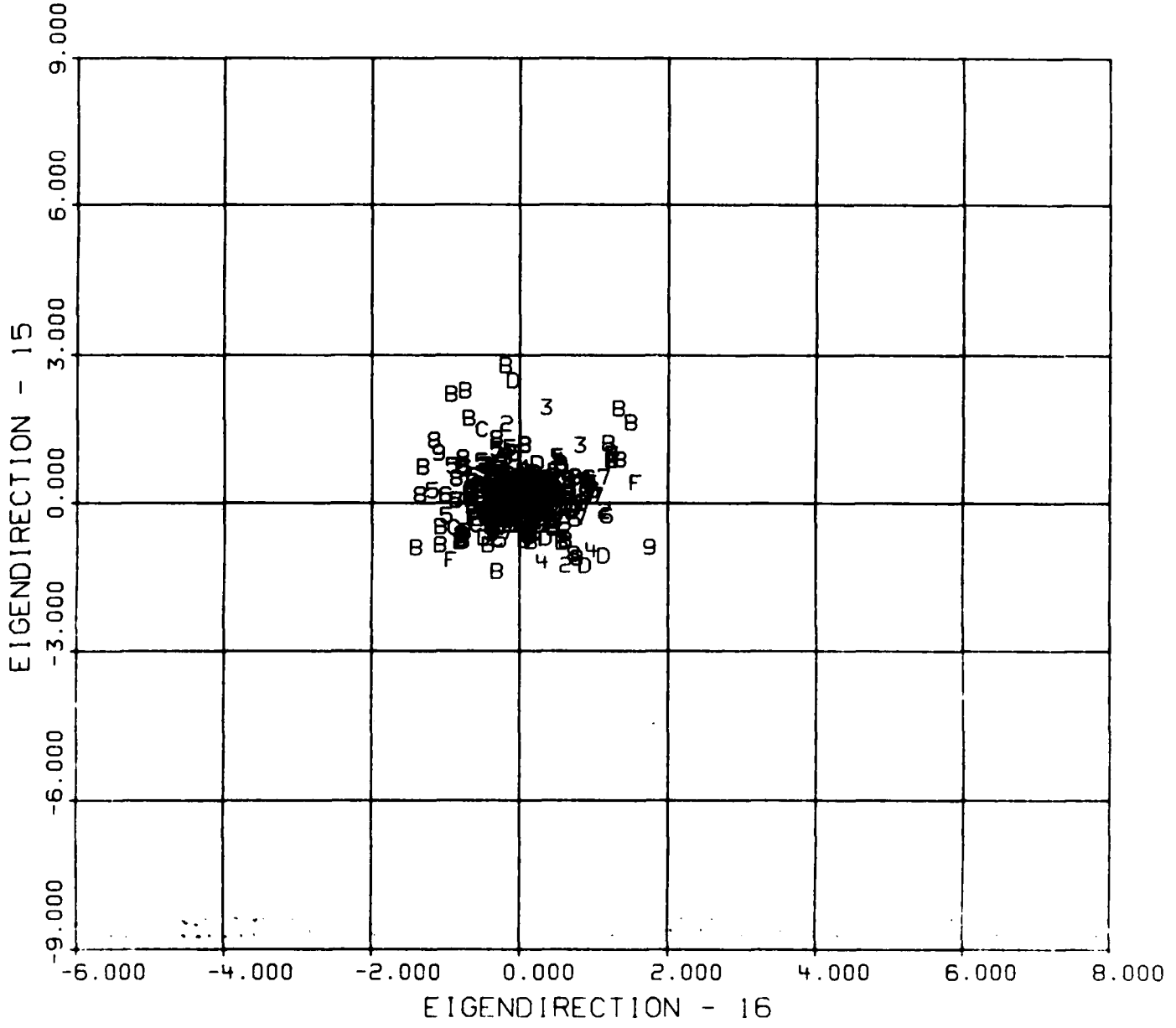


FIGURE-26 EIGENVECTORS 17-20

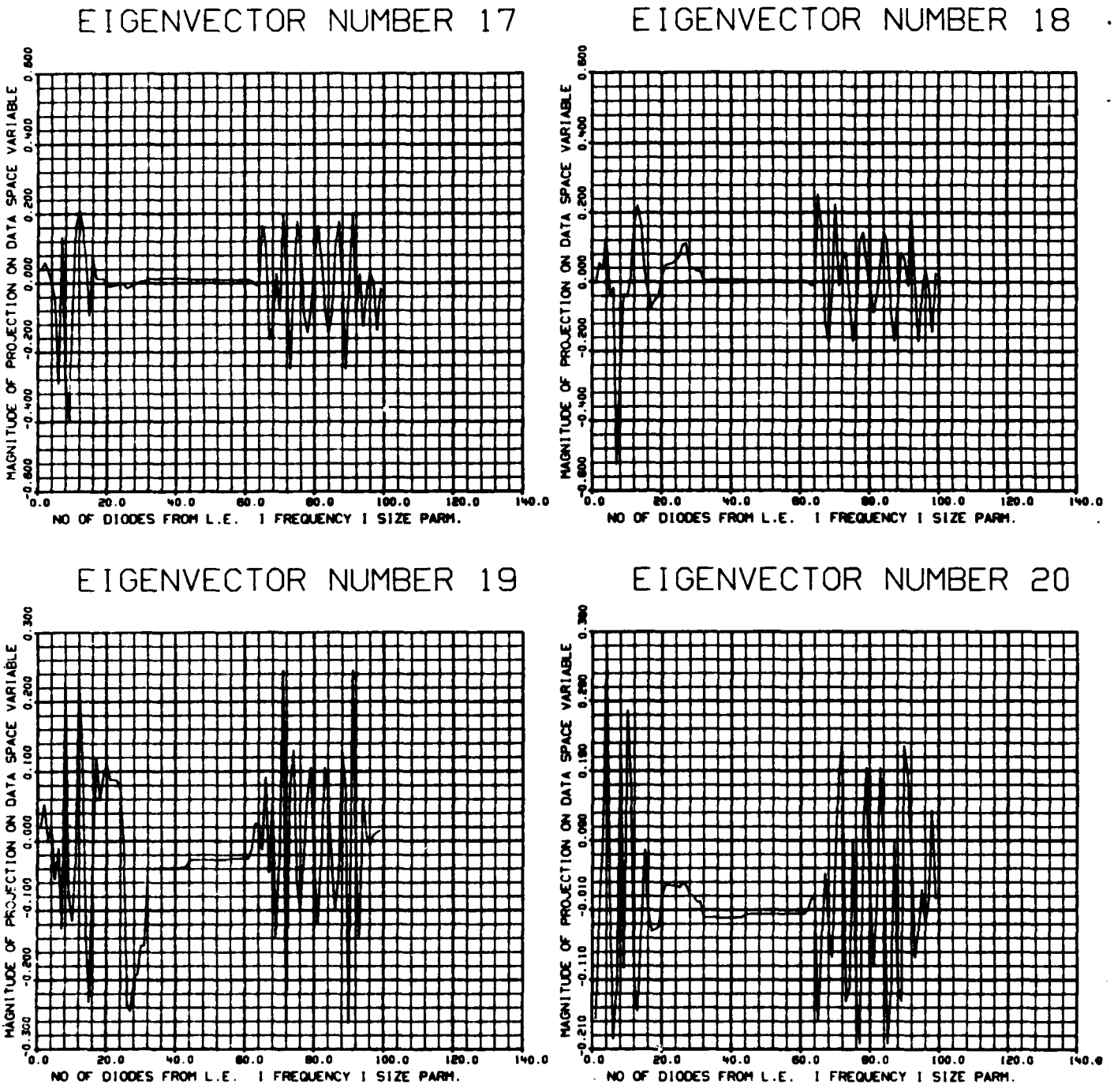


FIGURE-27

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 17 AND 18 - (1,A=NDL; 2,B=DNDRT; 3,C=BLT; 4,D=COL; 5=BLT OR COL; 6,F=PLT; 7=PLT OR SPH; 8=MISC; 9=STRKR)

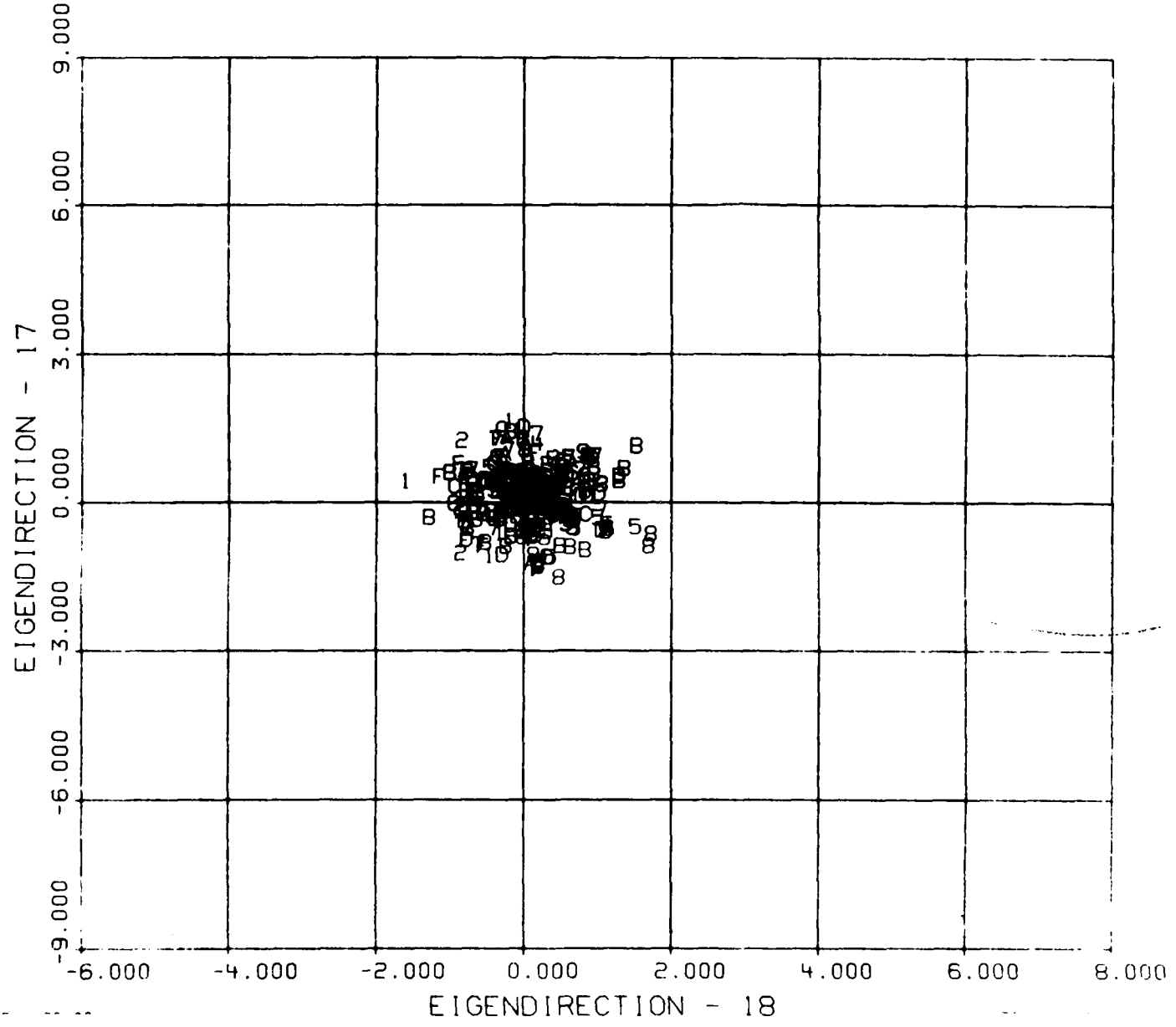
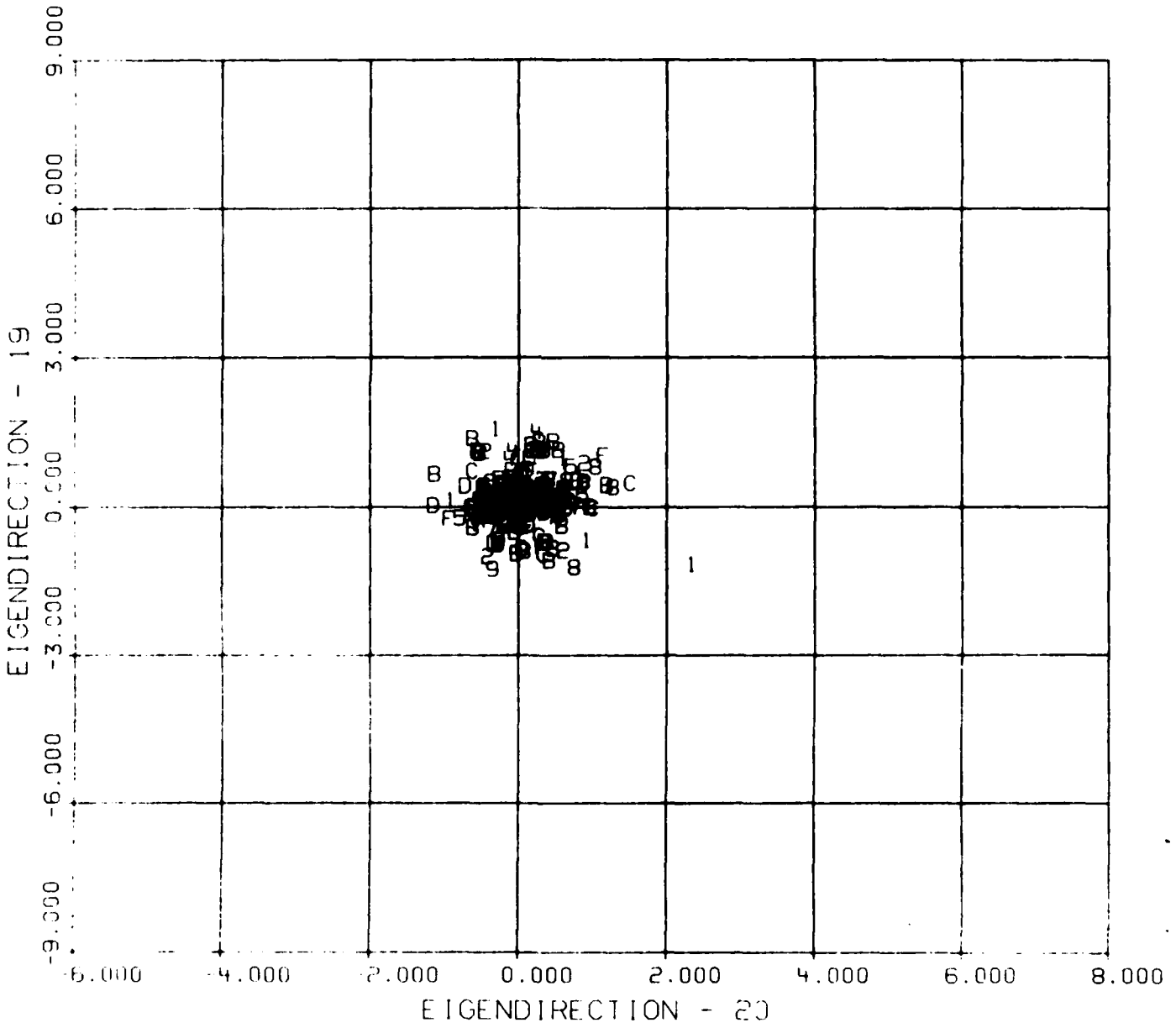


FIGURE-28

PROJECTION OF 403 CLOUD PARTICLES ONTO EIGENDIRECTIONS 19 AND 20 - (1,A=NDL; 2,B=DNDRT; 3,C=BLT; 4,D=COL; 5=BLT OR COL; 6,F=PLT; 7=PLT OR SPH; 8=MISC; 9=STRKR)



6.0 ANALYSIS OF RELATIVE IMPORTANCE VECTORS

In the description of the ADAPT approach presented in Appendix 1 and several of the references, it is pointed out that after the algorithm is obtained in the eigenvector space it may be transformed back to the original measurement space. In the measurement space, we may consider the plot of the algorithm a relative importance vector since the algorithm is a dot product. This means that the detection statistic is obtained by multiplying each component of the relative importance vector by the corresponding component of the original data vector. Thus, when the component of the relative importance vector is small, the corresponding component of the data vector makes a very small contribution to the detection statistic and conversely when the component of the relative importance vector is large, the corresponding component of the data vectors is very important to the detection statistic. Figures 29 through 34 present the relative importance vectors for the detection of: 1) dendrites, 2) needles, 3) columns, 4) plates, 5) streakers and 6) columns and needles. These are the six algorithms which have been incorporated in the one step and two step procedures which have been delivered as part of this contract.

The above description of how these relative importance vectors are used implies that if one wishes to compare two regions of the relative importance vector, that region which has the greatest area under the absolute value of the curve is the more important. Keeping this in mind, we note that in general, both the spacial components (i.e. components 0 through 64) and the frequency components (65 through 94) make approximately equal contributions to the decision and the size parameters a relatively small contribution. The one major exception to this is the streaker algorithm where the frequency and spacial components makes considerably smaller contributions relative to the size parameters. This is probably in part due to the fact that this algorithm was derived by manual examination of the projection of the three streakers which were available for the training on only the first two eigendirections. It suggests that significant improvements in the streaker algorithm would be possible if a relatively large number of streakers were processed through the Fisher classifier and a higher dimensional analysis performed to derive this algorithm. However, as pointed out in the introduction, the classification of streakers was not considered a major problem area and, therefore, little emphasis was placed on the development of this algorithm.

FIGURE -29 RELATIVE IMPORTANCE VECTOR FOR DETECTION OF DENDRITES

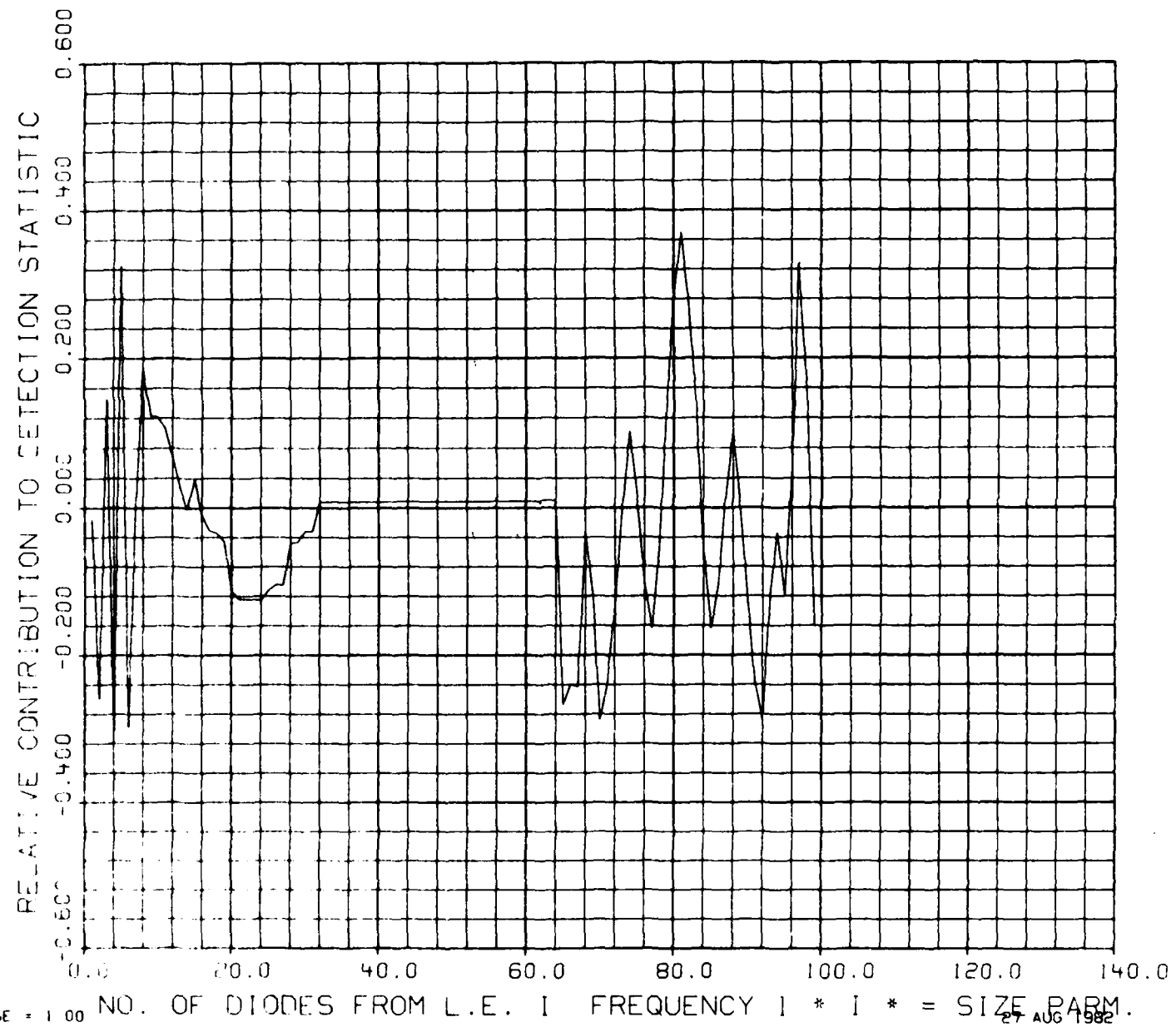


FIGURE -30 RELATIVE IMPORTANCE VECTOR FOR DETECTION OF NEEDLES

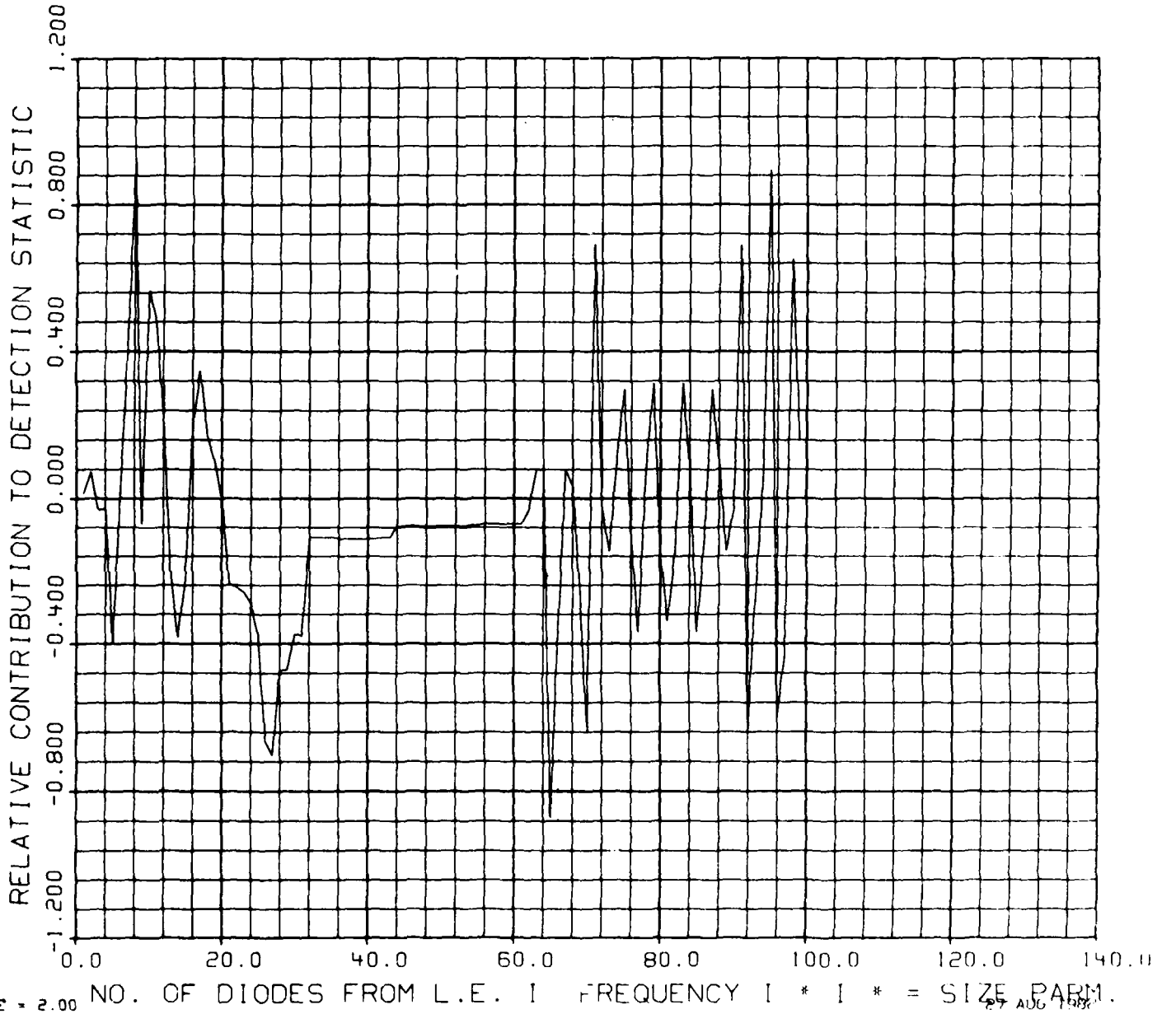


FIGURE -31 RELATIVE
IMPORTANCE VECTOR FOR DETECTION OF COLUMNS

FIGURE -32 RELATIVE IMPORTANCE VECTOR FOR DETECTION OF PLATES

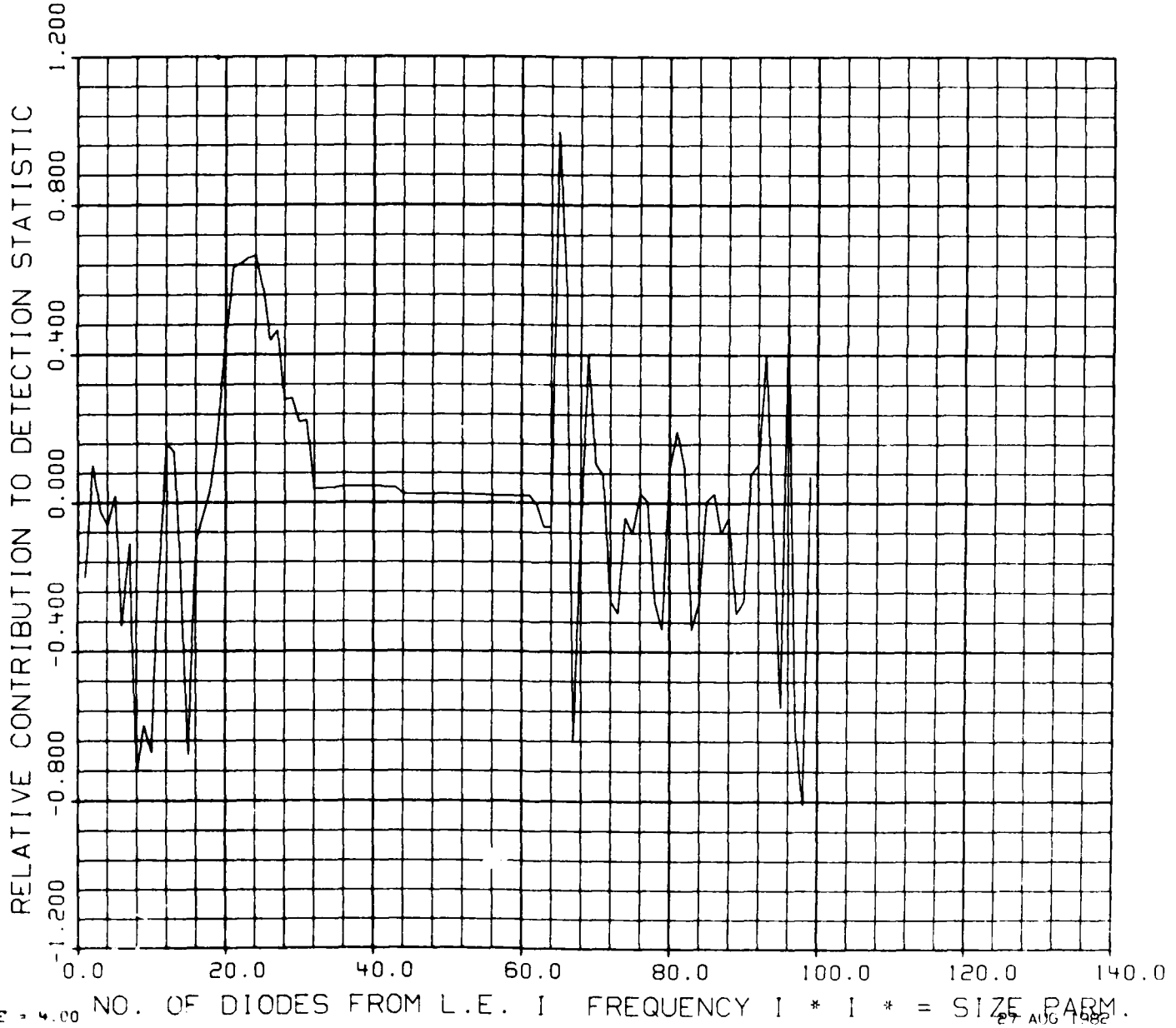


FIGURE -33 RELATIVE IMPORTANCE VECTOR FOR DETECTION OF STREAKERS

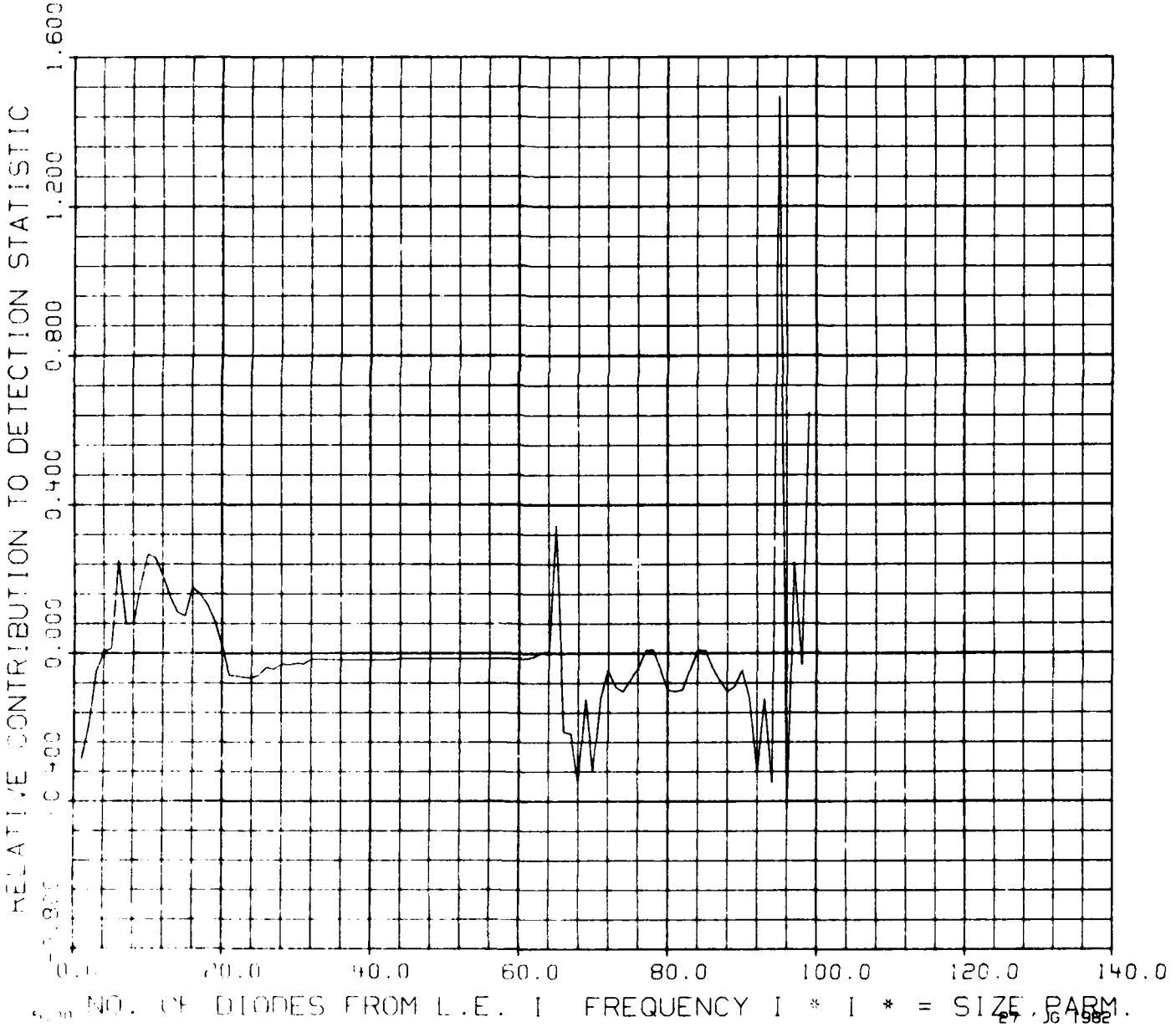
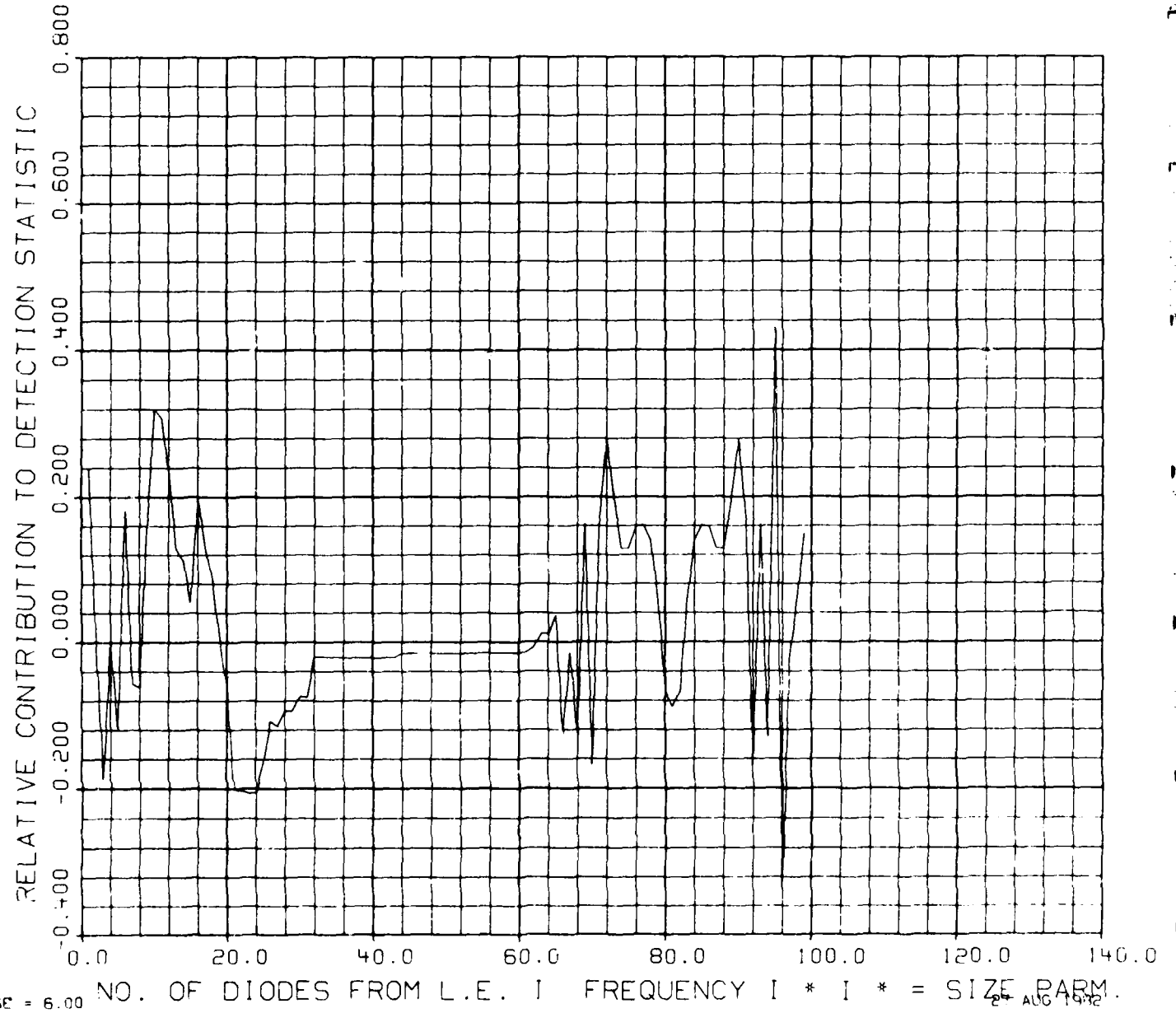


FIGURE -34 RELATIVE IMPORTANCE
VECTOR FOR DETECTION OF COLUMNS AND NEEDLES



In general, one observes the expected behavior in both the spacial and frequency components. That is, all of the objects tend to provide a similar return very near the beginning and then as one expects, the larger objects provide more return near the rear portion. The longest objects in the training data set were very large columns and plates which appeared in some of the data. For this reason, we find positive values of the relative importance vectors after approximately 20 diodes for both the columns and plates and negative values for the dendrites and needles which tended to be shorter than 20 diodes for most of the training data. When one examines the frequency space, one notes that long constant objects tend to have a higher frequency return then highly variable objects. This corresponds to the well known principal that if the time domain has a very smooth function extending over all of the space, one will have almost all of the frequencies represented in the frequency space and conversely a highly oscillating time function will tend to have a relatively smooth representation in the frequency space. It is interesting to notice that the columns and needles tend to cancel each others peaks and valleys such that the detector to detect the combination of columns and needles results in a very broadband signal in the frequency space. This is probably the explanation as to why this two step algorithm shows advantages over the single step of finding the columns and needles independently.

7.0 CONCLUSIONS

Two schema for the machine classification of cloud particles as measured by Knollenberg 2D probes have been developed and analyzed. These schema are based on detection and classification algorithms derived from real data and have incorporated into them the flexibility to allow the user to modify critical thresholds associated with these algorithms so that they may be adjusted to the needs of the user as well as to differences between characteristics of data sets. Examples have been given which show how the confusion matrices associated with a given training subset of a data set can be used to select and verify these adjustments to the constants.

The nominal performance of these schema using the nominal settings for the thresholds has been found to be superior to the performance that would be expected from manual classification. Thus, even without these adjustments these schema should provide performances superior to manual classifications. Since manual classifications is an extremely time consuming job and there is a tremendous number of particles in a typical data set, the advantages of these schema over manual classification are clearly significant even without the adjustment of the schema to a particular data set.

The techniques differ from those previously developed primarily by the use of real data for training data. Analysis presented show that even the variation between real sets of data can be expected to have significant effects on the proper design of the classification schema. Thus, it is very unlikely that the use of artificial data will yield algorithms which are useful for automated classification of these Knollenberg 2-dimensional cloud particles.

The classification schema developed here make use of the relatively simple to apply Fisher classification scheme rather than the more complicated maximum likelihood and Bayes family c classifiers. Considering the problems associated with the definition of good training sets and consistency of data between data sets, it is believed that the difference between the Fisher classifier which is an approximation to the maximum likelihood are small compared to the other problems associated with the development of these classifiers. Thus, the significant savings in computation both in the development of the classifiers and especially in the application of the classifiers justifies the use of these computationally more efficient algorithms.

The algorithms efficiency is further increased by the use of the unique capabilities of the ADAPT family of classifier development programs to objectively define and extract features from large data sets. Analysis of both rotated and unrotated data has demonstrated that the features used are relatively insensitive to particle orientation.

Initial attempts to reach agreement on the truth data suggested that the particles to be classified could be considered "poorly defined" objects. These objects were poorly defined in the sense that there will be disagreement as to proper classification for significant percentage of the members of the classes among different human classifiers. To understand the impact of this poorly defined nature of the objects, comparisons were made between 15 different human classifications of the same set. It was found that technician and computer specialists were, in general, more consistent with the trainer's classifications then experienced meteorologists. The particles causing the greatest difficulty for machine classification were also found to be different from the particles which caused the greatest difficulties for manual classification.

Analysis of the eigenvector expansion suggested that useful information may be found in any of the first 20 eigenvectors. The first eigenvector was primarily related to the size of the particle, the second eigenvector was primarily determined by the length, the third through 20th eigenvectors all showed significant contributions from both the spacial and frequency portions of the spectra. Analysis of the relative importance vectors shows that except for the streakers, both the spacial domain and the frequency domain were important to the classification. The one universal characteristic of all of these algorithms was the symmetry of the frequency plane relative importance vector about the mid point. The dominant features of the time and frequency domain could both be explained in terms of the shape of the particle. The streaker classification was based primarily on the global magnitude and shape parameters with less contribution from either time domain or the frequency domain of the signature.

REFERENCES

1) Rahman, Mizanur, M., et. al.; "2-D Hydrometeor Image Classification by Statistical Pattern Recognition Algorithm, J. Appl. Meteor. Vol 20, No. 5 pp 536-546, 1981

2) Rahman, Mizanur, M., et. al.; "Feature Extraction and Selection for Pattern Recognition of 2-D Hydrometer Images, J. Appl. Meteor. Vol. 20, No. 5 pp 521-535 ,1981

3) Hunter, Herbert E., Dyer, Rosemary M., and Glass, Mort; "Comparison of Human and Machine Classification of Poorly Defined Patterns", to be submitted to IEEE-PAMI

4) Knollenberg, R. (1970) The optical array: an alternative to scattering, or extinction for airbornes particle size determination, J. Appl. Meteorol. 9:86-103

5) Knollenberg, R. (1975) The Response of Optical Array Spectrometers to Ice and Snow: A Study of Probe Size to Crystal Mass Relationships, AFGL-TR-75-0494, ADA020276

6) Knollenberg, R. (1976) The Response of Optical Array Spectrometers to Ice and Snow: A Study of 2-D Probe Area-to-Mass Relationships, AFGL-TR-76-0273, AD A034 741

7) Heymsfield, A.J., and Knollenberg, R.G. (1972) Properties of cirrus generating cells, J. Atmos. Sci. 29:1358-1366

8) Heymsfield, A.J. (1976) Utilization of aircraft size spectra measurements and simultaneous Doppler radar measurements to determine the physical structure of clouds, J. Atmos, Sci. 29:1358-1366

9) Cunningham, R.M. (1978) Analysis of particle spectra data for optical array (PMS) 1D and 2D sensors, Preprints, 4th Symposium on Meteorol, Observations, pp. 345-350.

10) Hunter, Herbert E., et al; "An Objective Method for Forecasting Tropical Cyclone Intensity Using NIMBUS-5 Electrically Scanning Microwave Radiometer Measurements" JAM, Vol 20, No 2, Feb 1981

11) Lackenbrach, P.A. and Mickey, M.R.; Estimation of Error Rates in Discriminant Analysis "; Technometrics, 10, 11-17, 1968

12) Preisendorfer, R.W. and Barnett, T.P., 1977; "Significance Tests for Empirical Orthogonal Functions", Reprints 5th Conference Prob. & Statistics in Atmospheric Science, Amer. Met. Soc. 169-72

13) Preisendorfer, R.W. and Overland, James E.; " A Significance Test for Principal Components Applied to Cyclone Climatology" Mon. Wea. Rev., Vol 110, No-1, pg 1

14) Anderson, T. W. and Bahadur; "Classification Into Two Multivariate Normal Distribution with Different Covariance Matrices", Annually of Math. Stat., Vol 33 pg 420-431, 1962

APPENDIX A

REVIEW OF ADAPT APPROACH TO EMPIRICAL DATA ANALYSIS

SEPTEMBER 1982

This attachment will present the detail information which defines the ADAPT approach to empirical data analysis. This approach is based on the concept that empirical data analysis should be preceded by transforming the data from the original data space to a more efficient analysis space. This more efficient analysis space is defined as that space which requires the least number of numbers to represent a given amount of information in the original data set. It can be shown that this space is simply the eigenvector space and the transformation required is the eigenvector or the Karhunen-Loeve transformation.

The personnel who are now the senior technical staff of the ADAPT Service Corporation each have a decades experience with analysis in the eigenvector space. This has led to the development of a unique set of computer programs both to perform the transformation to the eigenvector space and to perform the analysis in this space.

The ADAPT programs have many outputs which are considerably different from those which are obtained from classical approaches to empirical or statistical analysis. This attachment will attempt to present a brief description of these outputs and how they may be used to improve empirical data analysis. In the following paragraphs, we will summarize each of the capabilities and outputs of the ADAPT analysis procedure.

ADAPT OPTIMAL REPRESENTATION

The major difference between the ADAPT approach to empirical analysis and the classical approach to empirical analysis is the derivation and use of the ADAPT optimal representation to simplify and improve all subsequent empirical analysis of the data. The ADAPT optimal representation is known in the literature under the names of: 1) principal component analysis, 2) Karhunen-Loeve expansion, 3) eigenfunction expansion and 4) optimum empirical orthogonal functions. The ADAPT Service Corporation has developed a unique approach to obtaining this transformation which overcomes the difficulties associated with the iterative techniques discussed in the literature and available in most "statistical packages". The importance of this unique approach to deriving eigenvectors is discussed in the ADAPT write-up titled "significance of ADAPT Approach to Deriving Eigenvectors" included as Appendix 2B.

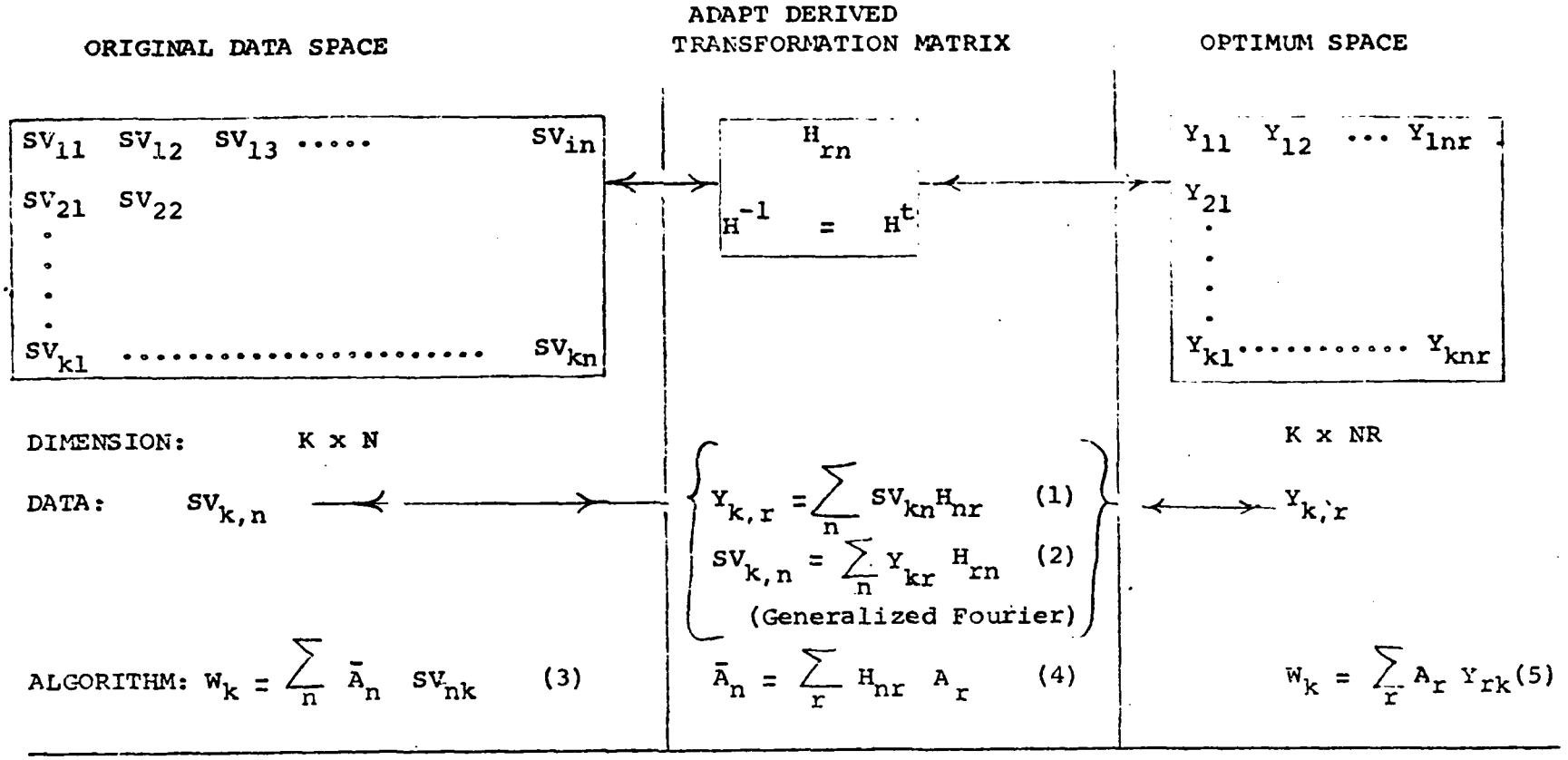
It is useful to review some of the basic concepts associated with the ADAPT optimal representation. The first point which must be established is the meaning of optimal. For the ADAPT application, optimal is defined as that representation which requires the least number of numbers to represent a given amount of information or variation. Thus, by definition, the ADAPT optimal representation

is the most efficient orthogonal coordinate system for representing the learning data. For further discussion of this transformation and its use see References 1-3.

After the optimal representation has been obtained, the learning data is transformed to the optimal space and the analysis is performed in the optimal space. The optimal space may be viewed either geometrically as a new coordinate system for describing the learning data of functionally as a system of empirical orthogonal functions (EDF) to be used to construct a generalized Fourier series representation of the learning data. In the first case, the analysis is performed on the coefficients of each of the data vectors in the new space. In the latter case, the analysis is performed on the coefficients of the generalized Fourier series expansion of each of the data histories. The numerical value of the coefficients is identical regardless of whether the procedure is visualized geometrically or functionally. Thus, the major output of the first step of any ADAPT analysis is the transformation matrix to transform the data from the original data space in which the data vectors are defined to the new optimal ADAPT analysis space.

To visualize the role of the ADAPT representation, consider the transformation matrix H_n between the original data space containing observation or data vectors "SV_{kn}" and the optimal analysis space containing the transformed data vectors "Y_{kr}". Figure 1 presents a block diagram of the ADAPT process illustrating this role of the optimal space. The transformation matrix, H_n, is an orthogonal matrix, the inverse of this transformation is equal to its transpose. Thus, one has rules for transforming the data to the optimal space and the results of the analysis from the optimal space back to the original space. The dimensionality of the original space which using the notation of Figure 1 is K by n will be reduced to K by r where K is the number of cases and n is the number of dimensions or number of numbers required to describe each case. In general, for large data sets, r is at least an order of magnitude less than n. For data vectors less than the order of one hundred, r may only be a factor of 2 to 10 less than n. The data in the original data space is designated by the symbol V_{kn}. In the optimal space, his data is represented by the coefficients Y_{kr}. Where K in both cases designates the case and n and r designate the components of the data vector in each of the spaces, respectively. One may transform the

FIGURE -1 - BLOCK DIAGRAM OF ADAPT PROCESS



THUS: r-th OPTIMAL FUNCTION = $H_{r,n}$

RELATIVE IMPORTANCE VECTOR = $\bar{A}_n$

AND: W_k = THE DETECTION STATISTIC OR ESTIMATE

RELATIVE IMPORTANCE SPECTRA = A_r

SCATTER PLOT Y_{1k} vs Y_{2k}

data either from the original data space to the optimal space or visa versa by using the transformation matrix as indicated by the arrows on Figure 1. Linear algorithms may also be transformed between the data space and the optimal space by use of the H matrix.

The ADAPT characteristics which in addition to the classical statistical summary parameters would be of interest include the ADAPT optimum function, the information energy plot, the ADAPT scatter plot, the ADAPT relative importance vector, performance map, independent eigenscreening and the empirical validity criteria. The following paragraphs will present a brief description of each of these and some of the ADAPT preprocessing concepts.

Optimum Function

Referring to the preceding descrption of the ADAPT process, the ADAPT optimum function is numerically the corresponding column of the H matrix. Since this vector is described by N components, it has the appearance of a data vector that shows the importance of each of the original components of the data vectors to the construction of the optimal space. Plots of this function provide a physical interpretation for the components of the optimal space and also an indication of which of the original data vector components are conveying similar information. This may be viewed as an analysis of variation of the data but it should not be confused with the classical analysis of variation which is normally associated with the outputs of regression analysis. These classical analyses of variation generally describe how much of the variation observed in the dependent variable can be explained by the independent variable. The ADAPT optimal functions on the other hand, are simply an analysis of variation of the independent variable without considering the dependent variable at all. It seeks to answer the question which independent variables express the greatest amount of variation and which independent variables convey similar information.

Information Energy Plot

The eigenvalues associated with each of the optimal

functions defines the amount of variation in the learning data set which is explained by that optimal function. Since all information must be conveyed by variation in the data, this variation is analogous to an "information energy". One of the standard ADAPT outputs which will be provided for each of the bases developed (i.e. transformation matrices) in this study is a plot of this information energy or eigenvalue as a function of a number of dimensions used. Examination of this information energy curve allows one to determine the dimensionality at which the information has the character of noise. One can also observe the change of character of the information represented as a function of dimensionality. It is often possible to detect the point at which the eigenvectors are primarily correcting for anomolous cases! Thus, the information energy is <u>one</u> of several important tools in selecting the dimensionality for the analysis. Some of the subtle aspects associated with analysis of the information energy are discussed in references 4-6. Ref-4 is a fundamental paper, the results are often misused. Ref-6 discusses this in more detail.

Scatter Plot

The ADAPT scatter plot is the projection of the data vectors under consideration on two dimensions of the optimum space. In general, one projects on the first two dimensions on the optimal space since these two dimensions provide the best representation of the information contained in the data. This is identical to making a scatter plot of the Y_{1k} versus Y_{2k} coefficients. Note, that equation 2 on Figure 1 can be interpreted as the generalized Fourier series expansion of data history SV_k in terms of the orthogonal functions defined by the H matrix. Thus, a data history, SV_{pn} having a first coefficient of 1, ($Y_{1p} = 1$), on the scatter plot and a second coefficient of -1, ($Y_{2p} = -1$), would have a two term generalized Fourier series representation equal to the difference between the first and second optimal functions ($SV_p = H_1 - H_2 \quad ...$). The significant achievement of the scatter plot of the first two optimal coefficients of each of the data vectors is that it presents the best possible two dimensional representation of the entire data set. Each point on the scatter plot represents an entire history made up of N points.

Algorithm and Relative Importance Vectors

The derivation of a linear classification algorithm may be looked upon as the search for the line or vector with the property that the numerical value of a data vector's projection on this line is a good detection statistic. The ADAPT algorithm vector is a plot of the components of the projection of this vector in the original data space. Since the dot product of this vector with the data vector determines the detection statistic, the magnitude of each of these components provides a measure of the importance of each component to the algorithm being evaluated. In the ADAPT programs, the algorithm vector is derived in the optimal space. Thus in data space this vector is the product of the vector defined in the optimal space, A_I times the transformation matrix Π_{nr}.

The importance of any variable to an algorithm is the product of two values: 1) the value of the algorithm associated with that value and 2) the amount of variation associated with the variable. For example, a given variable makes no contribution to an algorithm if the algorithm value is zero or if it has the same value for all observations. Thus, we define the relative importance vector as a vector in data space where each component is the product of the algorithm value and the variance of the variable associated with that component. It follows from the mechanism of the dot product operation that it is the absolute value of the relative importance (or algorithm) vector which is significant. For example, considering the algorithm vector, if one variable has a value of minus .5 and another variable a value of plus 0.1 a change in the indexing variable having the value of minus 0.5 in the algorithm vector has five times the effect on the answer or detection statistic as the same change in the indexing variable having a value of plus 0.1.

Performance Map

The performance map is a plot of the dimensionality used for the analysis versus the performance of the algorithm developed. It provides an empirical non-parametric tool to determine whether there was sufficient learning cases

to provide a physically meaningful algorithm. It also provides a tool for estimating the gains possible by increasing the amount of training data. The task accomplished is analogous to the problem of fitting a third order polynomial to independent test data. One can always fit a third order polynominal to three numbers, and there is no implied physical significance to the fact that there is a good fit. This is often referred to as an overdetermined problem. On the other hand, if one has a "large" number of independent samples say one hundred samples and one fits a curve to this larger set of samples, one may conclude that those hundred samples can be approximated by a third order polynomial expression over the range of the available experimental data.

The question is, what is "large"? The same phenomena occurs for all empirical analysis. If the number of learning cases equals the number of dimensions, most empirical algorithms will fit the learning data exactly, however, once again there is no physics implied in this fit. As one increases the number of learning cases beyond this point, if one continues to achieve good fits of the data with the empirical algorithm, the probability that the fit is based on physics increases. Eventually when the ratio of learning cases to number of dimensions used is "sufficiently large", one not only can assume that the relationship is based on physics but that the performance which is obtained on the learning data may safely be extrapolated to future independent test samples. The ADAPT performance map can be used to define "sufficient large".

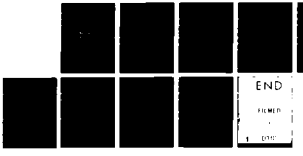
After introduction of the independent eigenscreening concept into the ADAPT linear classification and regression programs, the performance map was no longer required to determine if the overdetermined situation had been obtained. However, the performance maps are now easier to use and still determine if additional training data should be used. They now provide plots of both the biased and unbiased performance as a function of ratio of number cases to dimensionality. When both the biased and unbiased performances are similar, the number of training cases are adequate for that algorithm.

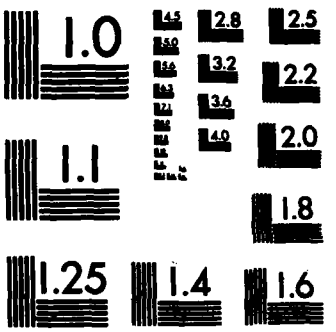
Empirical Validity Criteria

The ADAPT approach of preceding the [illegible] with an optimal representation also provid[illegible]

AD-A123 402 MACHINE CLASSIFICATION OF CLOUD PARTICLE TYPES(U) ADAPT 2/2
SERVICE CORP READING MA H E HUNTER AUG 82 ADAPT-82-4
AFGL-TR-82-0298 F19628-81-C-0047

UNCLASSIFIED F/G 4/2 NL





for performing a necessary but not sufficient test to determine whether an empirical algorithm is applicable to a new independent test sample. This empirical validity criteria consists of obtaining the ratio, (Q) of the length of the data vector for the new independent test case in the optimal space to its length in the original data space. If this ratio is significantly less than the corresponding ratio for the average or typical learning data used to derive the algorithms, the independent test case has been obtained from a sample which is significantly different from the learning data. Thus, empirical analysis of the test case based on algorithms derived from that learning set can not be justified. Experience with this validity criteria in many different problems, has shown it to be very effective in providing apriori estimate of whether an algorithm is applicable to a particular test case. This procedure has been part of the ADAPT family of computer programs and was first described in the literature in Ref-7.

Group-Out Independent Testing

The ADAPT regression and classification algorithm development programs include a capability to obtain independent (i.e. unbiased) test results with a minimum increase in the required number of cases. This is achieved through the group-out testing procedure. The procedure is to consider the original training set of data as made up of a relatively large number of small groups of cases. Note, that the group may be as small as one case. If we have a set of M cases available for the study and we use groups of N cases each, the procedure is to remove the first group of N cases giving a training set of M minus N cases and an independent test set of N cases. The algorithms are derived on the training set and tested against the N cases in the first group. When this is completed, the N cases in the first group are returned to the training set and a second group of N cases is removed and the procedure repeated. If this procedure is followed, one finds that they have derived a total of M divided by N algorithms each having M minus N training cases and has tested the total of all of these algorithms against M independent test cases. Thus, the net effect of this procedure is to effectively provide M minus N training cases and M independent test cases from a set of M cases.

It should be noted, that the procedure originally reported in the literature in Ref-8 is based on the capability to obtain an inverse with one case omitted from the covariance matrix. In the ADAPT programs, our ability to use this procedure is due to the efficiency of the analysis in the ADAPT space. Although we also have some programs which make use of the procedure outlined in the literature combined with the efficiency of the ADAPT analysis space which provides an extremely economical way of performing one-out testing. We have compared the performance obtained with the group-out testing with classical independent testing and have found with random selection of groups, stable sets of algorithms produce identical results as independent tests when training and test samples are drawn from homogeneous data. With conservative selection of groups the group-out testing is a more severe test.

Eigenscreening

Classical screening regression has been avoided in the development of the ADAPT computer programs for two reasons. These reasons are: 1) classical screening regression makes the screening decision based on the performance established from training data. Comparative analyses between use of independent test data and training data performed by the ADAPT Service Corporation have shown that the training data does not provide a reasonable basis for screening of the variables and 2) classical screening is performed on a set of independent variables which are not orthogonal and thus considerable effort is required ascertain whether a variable is retained because it is significant or because it is repeating information which has already been obtained in a different variable.

The ADAPT eigenscreening approach is similar to the classical screening regression except that the screening is performed in eigenvector space and performance is established based on the group-out testing procedure and thus is based on independent and unbiased test results. Since the screening is performed in the eigenvector space instead of the data space, the variables being screened are orthogonal and one need not be concerned with the linear dependence between the screened variables.

The screening process is significantly improved because: 1) the unbiased test provides a higher confidence performance estimate than dependent testing and 2) the group-out testing allows the evaluation of the stability of each term in the algorithm as well as the algorithm performance. The evaluation of the stability is especially important when the number of cases is limited, since the "overdetermined" solution, which must be avoided, is very unstable. If the performance of the different algorithms developed in the group-out testing is unstable, one can be certain that there are insufficient training cases. If any term in the algorithm is unstable, this term probably should have been rejected.

These improvements in the screening have resulted in significant additional capabilities in performing regression analysis. The ADAPT Service Corporation has also applied these procedures to pattern recognition techniques and has computer programs which provide these same advantages to the development of classification algorithms. Further discussion and examples illustrating the ADAPT eigenscreening are given in the ADAPT write-up "Illustration of ADAPT Independent Eigenscreening Technique", included as Appendix C3D.

Preprocessing

An extremely important factor in obtaining good empirical results is to preprocess the data such that the information is presented in a useful manner. The ADAPT family of computer programs include the capability to provide most of the classical preprocessing such as normalizations, adjusting the data according to some prescribed function, taking Fourier or cepstrum transforms of the data. The ADAPT computer programs also include specialized preprocessing which has been developed based on requirements established as a result of the work performed in the past. These include such techniques as equalization, thresholding and a unique capability for objectively deriving folding procedures to overcome non-linearities and non-monotonic relations between the predictant and the predictor variable.

The last preprocessing performed before processing the data through the ADAPT eigenvector derivation programs or transforming the data to the eigenspace is to reduce the data to zero mean by subtracting the average of all the training cases from each data vector. The zero mean data offers a great many numerical advantages and is used in almost all ADAPT studies.

In all studies where different types of variables (eg. a data vector composed of temperature measurements and pressure measurements) and many cases where the variables are the same but their magnitudes may mask the variation, the reduction to zero mean is preceded by "equalization" of the data vector. This is a process by which the value of each variable is forced to lie between 1.0 and 2.0. This is accomplished by transforming the original variable $V_k(x)$ to a new variable $\tilde{V}_k(x)$ using:

$$\tilde{V}_k(x) = 1 + \frac{V_k(x) - VMIN(x)}{VMAX(x) - VMIN(x)} \qquad (1)$$

where $V_k(x)$ = Value of observation k associated with index x

X = A range of one or more indexing variables

VMAX = Max value over all training data associated with index x

VMIN = Min value over all training data associated with index x

APPENDIX B - SIGNIFICANCE OF ADAPT APPROACH TO DERIVING EIGENVECTORS

INTRODUCTION

The ADAPT approach to empirical data analysis is that empirical data analysis such as pattern recognition or regression should be preceded by transforming all of the data into the appropriate eigenvector space for analysis. This provides an optimum (in the Karhunen-Loeve sense) space in which to perform the analysis and significantly decreases the cost and increases what can be learned from any subsequent analysis. This approach translates most of the major numerical analysis problems into the first step (i.e. finding the eigenvectors of the covariance matrix derived from the original data vectors). Thus, the efficient and correct derivation of the eigenvectors associated with a covariance matrix is one of the most important aspects of the ADAPT approach to empirical analysis.

The ADAPT Service Corporation uses a unique approach to the derivation of these eigenvectors which provides both a greater efficiency with respect to computer running time and core size and also eliminates the problems resulting from noisy and/or ill-conditioned real data sets. These noise and data conditioning problems are very similar to the problems which lead to singular matrices when analyzing data in the original data space. Although these problems do not cause a failure to obtain an answer with conventional eigenvector techniques such as those included in the IBM scientific sub-routine package they often lead to meaningless outputs from these techniques and unnecessarily large requirements for core size and running time. This appendix will review these difficulties and outline the advantages of circumventing these difficulties prior to entering the procedures for deriving eigenvalues and eigenvectors.

PITFALLS OF CONVENTIONAL EIGENVECTOR DERIVATIONS

Since we are dealing with the task of finding the eigenvectors of the covariance matrix, we may limit the discussion to real symmetric matrices. Modern techniques (i.e. the Jacobi technique which is used in the IBM scientific sub-routine package or the Givens-House-holder technique described in Reference 1 and used in many commercially available statistical packages are based

on iterative techniques which usually proceed from some initial guess for the eigenvalues and an apriori specified accuracy. With the judicial use of overflow and underflow protections in the programming of these techniques, one obtains a set of numbers and vectors which look like eigenvalues and eigenvectors. In many ways, this is unfortunate because unlike the situation with matrix inversion where ill-conditioned input data leads to the impossibility of obtaining an answer, ill-conditioned data leads to a partially incorrect answer with these eigenvector techniques. These incorrect outputs are responsible for many of the misconceptions concerning eigenvector analysis that are often heard and occasionally even appear in the literature regarding the use of eigenvectors as an analysis tool. The most common of these misconceptions are:

1) the instability of eigenvectors (i.e. cases where eigenvectors corresponding to relatively large eigenvalues are supposedly unstable as one changes the data slightly),

2) the statement that the derivation of eigenvectors for large real data vectors is nearly computionally impossible (zb: Reference 2, Page 31) and,

3) only the first few dominant eigenvectors can have physical meaning (the ADAPT Service Corporation has found and verified physically meaningful information in eigenvectors explaining considerably less than 1% of the variation).

In the following paragraphs, we will discuss two problems which may lead to such false conclusions, these are:

1) insufficient independent observations and,

2) noise.

The inpact of insufficient observations can be seen most clearly by considering a simple case. Suppose for example one had three observations (i.e. cases) of some phenomena where each observation consisted of five independent measurements associated with the phenomena being observed. This will provide a data matrix consisting of three vectors of five components each. Clearly, if one attempted to run a five dimensional regression or a discriminate analysis requiring the inversion of the covariance matrix in this five dimensional original data space, they would not be surprised to find that the matrix to be inverted is singular. Similarly, one should

not expect to be able to find five eigenvectors associated with this data set. Table 1 shows an example using the Givens-Householder technique where the substitution of the covariance matrix associated with these three five component data vectors into conventional eigenvector routines will lead to <u>five</u> eigenvalues and <u>five</u> associated eigenvectors. The two smallest eigenvalues and their associated eigenvectors must be meaningless and should be discarded.

TABLE 1 - SAMPLE VECTORS AND EIGENVECTORS DERIVED USING GIVENS-HOUSEHOLDER TECHNIQUE

3 - INPUT VECTORS

V1 =	-10.	0.	0.	0.	0.2
V2 =	0.	1.	0.	0.	0.
V3 =	10.	-1.	0.	0.	-0.2

5 - EIGENVECTORS AND CORRESPONDING EIGENVALUES:

	EIGENVECTORS					EIGENVALUE
E1 =	0.9985	-0.0503	-0.0200	-4.3E-12	0.0	200.
E2 =	0.0503	-0.9987	5.6E-8	-2.2E-17	-5.7E-17	1.5
E3 =	-2.8E-20	-7.0E-17	2.2E-10	-1.0	-1.0	6.0E-18
E4 =	0.	0.	0.	0.	1.0E-11	2.8E-16
E5 =	-0.0200	-0.0010	-0.9998	-2.2E-10	1.4E-15	0.0

If we now introduce two additional cases which are linearly dependent on the original three cases, there will be no charge in the above described situation except that in general the eigenvalues and the eigenvectors will change. However, if these two linearly dependent eigenvectors are noisy they may introduce additional positive eigenvalues, making the uses believe that there are more than three meaningful eigenvectors, even though a maximum of three eigenvectors can have any meaning. We would hope that these eigenvectors were those associated with the largest eigenvalues, however, this can not be assured. If each of the data vectors were similar such that the first eigenvector explained almost all of the variation and the second and third eigenvectors only explained a small amount of the variation, the eigenvalues of the noise generated eigenvectors may exceed the eigenvalues of the true eigenvectors. Thus, the ill-conditioned data which leads to most problems appearing as singular matrices in data space analysis when combined with noisy data will lead to the generation of false eigenvectors when one attempts to derive eigenvectors with most modern iterative techniques. Thus, the data conditioning necessary to insre successful results in the data space analysis is equally important to deriving the eigenvectors.

When dealing with real data especially with real data defined by a large number of observations especially where the number of cases is only slightly greater than the number of measurements defining each of the observations, linear dependence of cases within this data and noise may create these problems even where one would not expect them. Noisy data further aggravates the problem by decreasing the number of independent observations. Large sets of real data where each data vector is itself a high dimensional vector are particularly susceptable to a noise induced linear dependence. That is, although a given observation may in principal be independent of all other observations it may be sufficiently similar that the difference between it and another observation is within the noise or the inaccuracies of the measurements. When this occurs, it can dramatically decrease the number of independent cases available and it is often difficult to determine this effect apriori by examination of the data or even the physics of the process. Since this noise induced linear dependence will also reduce the total number of eigenvectors which can be expected from the covariance matrix, it's effect can appear in exactly the same way as the simple example given

above. Thus, we see that in dealing with real data, the use of poorly conditioned data in conventional eigenvector derivation procedures can lead to a large percentage of the eigenvectors being generated from measurement inaccuracies or other noise and having no real relationship to the data. Furthermore, this has been accomplished with a great deal of unnecessary effort on the part of the computer. This unnecessary effort has increased both the core size and running time required.

ADAPT EIGENVECTOR TECHNIQUE

The ADAPT technique to circumventing the conventional problems in deriving eigenvector representations is to precondition the data matrix by a proprietary procedure which eliminates the above described problems and is mathematically equivalent to orthogonalizing the matrix without optimizing. This preconditioned data is then used to derive the Karhunen-Loeve expansion appropriate to the original data.

REFERENCES

1. Wilkinson, J.H., "Householders Method for the Solution of the Algebraic Eigenproblem", Computer Journal, Vol. 3, Pg 23-27, 1960.

2. Andrews, Harry C., "Introduction to Mathematical Techniques and Pattern Recognition, John Walley and Son, 1972.

APPENDIX-C

ILLUSTRATION OF ADAPT INDEPENDENT EIGENSCREENING TECHNIQUE

Tables 1 through 3 present typical outputs from the ADAPT independent eigenscreening programs. These tables illustrate the development of a regression algorithm using independent eigenscreening for estimating the change in longitude of a tropical storm 24 hours after its observation. Before using these tables to illustrate the independent eigenscreening technique, we will describe the information presented on the tables. The tables consist of ten columns, each of these columns defines one parameter of interest.

To understand the information presented, we must recall that the procedure used is to divide the training cases into two groups, the first group to be used as training and the second group as independent test. For example, consider a set of 60 training cases, we might take the first 50 as the training and the last 10 as independent test. The algorith would then be derived using the first 50 cases and tested against the last 10. When this is completed, a different set of 50 training cases and 10 independent would be used. For example, we might now take Cases 1 through 40 and 51 through 60 as training data and test the results against Cases 41 through 50. After completion of the second set of algorithms and independent tests, we could repeat the procedure four more times. This would yield six different training algorithms and sets of 10- independent tests on each of the six algorithms for a total of 60 independent test cases. Thus, beginning with a total set of 60 cases this procedure would result in 50 cases for training and 60 independent test cases.

The selection of six sets of algorithms and the composition of each set are input parameters and are selected based on the physics of the problem. The penalty is that we need to develop six sets of algorithms. Using conventional techniques the cost of this procedure would be prohibitive for most real problems, but with the ADAPT procedures we can take this approach. As a result of this approach, we have a performance for the independent test cases, in this case, the correlation coefficient given in the third column of the tables and labeled RHOZVT. We also have a learning correlation coefficient for each of the six algorithms developed. The average of these

coefficients is given in the fourth column of the table and labeled RHOVL. We may also compute the standard deviation of this learning correlation coefficient and if the algorithm is stable we would expect that the standard deviation of the learning correlation coefficients would be small compared to the average learning correlation coefficient. Thus, we define the ratio of the standard deviation to the average correlation coefficient of the learning data as the learning stability. This is provided in the tenth or last column of the table under the title, "Learn Stab".

Since we have developed six algorithms and each algorithm has a number of terms in it equal to the dimensionality of the analysis given by Column 2 in the tables, we can also examine the stability of each term in the algorithm in the same way as we examine the stability of the performance. This stability is given in the tenth column of the table under the title of MAXSIG/MEAN. The value given is the value of the worst stability of any term in the algorithm, the number, NO, is given for some outputs and is the term in the algorithm which has this worst stability. When the stability exceeds an input threshold parameter, the entire stability for the algorithm is printed out (on a separate page from this summary table) so that the user may examine it. Our experience has shown that the learning stability is an almost certain test of having obtained the overdetermined solution. Experience with a number of different types of data and problems suggests that the stability parameter, MAXSIG/MEAN must be less than 0.5 to 0.7 for the last (ie. top) term.

The fifth column in the tables labeled, ACT-EST, gives the average error based on the independent testing. The three columns labeled, SDZV, SIGRATL, or SIGRZVDT, list the standard deviations and ratios of standard deviations which we have found useful in assisting in the understanding of the performance of the algorithms which have been developed.

The first column of Table 1 showing the potentially useful eigendirections provides a definition of which eigendirections are being used in any algorithms developed. In order to provide brevity in the table, only the last eigendirection added is listed. Thus, the bottom row of the first column of this table has a value: "3" , this indicates that the first eigendirection which was useful was the eigen-direction and that the algorithms developed to determine this

were developed in the one dimensional space (i.e. Column 2 headed NR has a value of NR = 1) consisting of the third eigendirection. The second row up shows that Column 1 contains the value "4". This indicates that the fourth eigendirection was the second useful eigendirection and the algorithm to determine this was derived in the two dimensional eigenvector space consisting of the third and fourth eigenvectors. Thus, if one were to read at the ninth row up from the bottom this is an algorithm developed in a 9 dimensional space consisting of all of the eigendirections shown in the first column from the ninth row down to the first row.

Now that we have looked at the format of Tables 1 through 3, we shall discuss their meaning. Table 1 summarizes all of the eigendirections which have been selected for retention based on the input parameters given by the user. That is, the user is allowed to input a criteria for both the independent test results and the stability which must be satisfied in order to retain a given eigendirection as a result of this screening. Table 2 shows the same results for those eigendirections which have been rejected based upon these criteria. In general, it is our practice on the first screening run to put in very weak constraints on the retention of eigendirections so that we retain any eigendirection which has any possibility of being useful in the analysis in this first pass. For this study, this yielded a total of 9 eigendirections which appeared to have some usefulness for the task at hand. We then repeat the screening procedure in reverse. That is, we start with all of the potentially useful eigendirections and sequentially delete one of the eigendirections and determine whether its deletion has improved or decreased the performance of the algorithm. It sometimes requires several passes. Table 3 presents the results of the last pass of this analysis. This table has the same format as Tables 1 and 2. In general, the criteria utilized are somewhat more stringent in these final steps. Examination of this table immediately shows the most successful algorithm, is the six dimensional algorithm using eigendirections 3,5, 7, 14, 8 and 9. Note, that at dimensionalities greater than or equal to six both the error and the stability

of the termsin the algorithm (MAX SIG/MEAN) deteriorate. This table then completes the screening process.

In summary, because of the efficiency of the ADAPT process, we have been allowed to make our decisions as to the value of retaining eigendirections based on independent tests as well as on the stability of the terms in the algorithm and the performance of the algorithm. Furthermore, we have not had to concern ourselves with the possibility that a given direction is being retained because of linear dependence on another eigendirection because of the orthogonal properties of the eigendirections. Although this example has been given for a regression analysis, our programs are completely operational and provide exactly the same results using similar outputs for a Fisher classifier. Similar procedures can be prepared for any linear classifier.

TABLE - 1

SUMMARY OF STORM OUT TESTING RUN ON 30 OCT 1982 (NR'S KEPT)

CLIPPER ONLY (LONGITUDE BIASED) FORECAST OF CHANGE IN 12HR LATTITUDE
RUN 20

LAST ACT-AVGL = 0.3538E 02

LAST STDDEV ACT = 0.4464E 02

ORIG NR	NR	RHOZVT	RHOVL	ACT-EST	SDZV	SIGRATL	SIGRZVBT	MAX SIG/MEAN NO.	MAX SIG/MEAN VALUE	LEARN STAB
14	9	0.7824E 00	0.8184E 00	0.2144E 02	0.2789E 02	0.5740E 00	0.6249E 00	2	0.965E 00	0.3928E-01
12	8	0.7500E 00	0.7920E 00	0.2266E 02	0.2967E 02	0.6101E 00	0.6648E 00	2	0.872E 00	0.3067E-01
9	7	0.7462E 00	0.7804E 00	0.2245E 02	0.2983E 02	0.6251E 00	0.6682E 00	2	0.656E 00	0.2111E-01
8	6	0.7403E 00	0.7659E 00	0.2304E 02	0.3004E 02	0.6428E 00	0.6730E 00	2	0.543E 00	0.2014E-01
7	5	0.7247E 00	0.7528E 00	0.2327E 02	0.3080E 02	0.6581E 00	0.6899E 00	4	0.380E 00	0.1824E-01
6	4	0.6585E 00	0.6933E 00	0.2639E 02	0.3366E 02	0.7203E 00	0.7540E 00	2	0.279E 01	0.2192E-01
5	3	0.6582E 00	0.6860E 00	0.2676E 02	0.3364E 02	0.7273E 00	0.7535E 00	2	0.318E 01	0.2120E-01
4	2	0.3195E 00	0.3632E 00	0.3219E 02	0.4236E 02	0.9304E 00	0.9489E 00	2	0.160E 00	0.1912E-01
3	1	0.2884E 00	0.3367E 00	0.3305E 02	0.4283E 02	0.9404E 00	0.9595E 00	1	0.164E 00	0.1665E-01

TABLE-2

SUMMARY OF STORM OUT TESTING RUN ON 30 OCT 1982 (NR'S DROPT)

CLIPPER ONLY (LONGITUDE BIASED) FORECAST OF CHANGE IN 12HR LATTITUDE
RUN 20

LAST ACT-AVGL = 0.3538E 02

LAST STDDEV ACT = 0.4464E 02

ORIG NR	NR	RHOZVT	RHOVL	ACT-EST	SDZV	SIGRATL	SIGRZVBT	MAX SIG/MEAN	LEARN STAB
13	9	0.7466E 00	0.7929E 00	0.2280E 02	0.2986E 02	0.6090E 00	0.6688E 00	0.8181E 00	0.3014E-01
11	8	0.7222E 00	0.7814E 00	0.2334E 02	0.3134E 02	0.6238E 00	0.7021E 00	0.1734E 02	0.2118E-01
10	8	0.7448E 00	0.7863E 00	0.2271E 02	0.2990E 02	0.6176E 00	0.6699E 00	0.3548E 00	0.2261E-01
2	1	-0.1401E 00	0.3451E-01	0.3559E 02	0.4488E 02	0.9991E 00	0.1005E 01	0.1334E 01	0.1098E-02
1	1	-0.1089E 00	0.6498E-01	0.3596E 02	0.4535E 02	0.9969E 00	0.1016E 01	0.9711E 00	0.2957E-02

TAPLE - 3

SUMMARY OF STORM OUT TESTING RUN ON 2 NOV 1982

CLIPPER ONLY (LONGITUDE BIASED) FORECAST OF CHANGE IN 12HR LATTITUDE
RUN 20

LAST ACT-AVGL = 0.3538E 02

LAST STDDEV ACT = 0.4464E 02

ORIG NR	NR	RHOZVT	RHOVL	ACT-EST	SDZV	MAX SIG/MEAN NO.	MAX SIG/MEAN VALUE	LEARN STAB	SIGRATL	SIGRZVBT
4	7	0.7886E 00	0.8108E 00	0.2106E 02	0.2747E 02	2	0.642E 00	0.3471E-01	0.5847E 00	0.6154E 00
9	6	0.7901E 00	0.8101E 00	0.2111E 02	0.2738E 02	5	0.150E 00	0.3579E-01	0.5858E 00	0.6134E 00
8	5	0.7740E 00	0.7963E 00	0.2199E 02	0.2827E 02	4	0.273E 00	0.3006E-01	0.6045E 00	0.6334E 00
14	4	0.7622E 00	0.7833E 00	0.2217E 02	0.2890E 02	4	0.163E 00	0.3653E-01	0.6210E 00	0.6475E 00
7	3	0.7274E 00	0.7455E 00	0.2369E 02	0.3064E 02	1	0.943E-01	0.2046E-01	0.6663E 00	0.6865E 00
5	2	0.6658E 00	0.6853E 00	0.2650E 02	0.3332E 02	1	0.123E 00	0.2075E-01	0.7280E 00	0.7464E 00
3	1	0.2884E 00	0.3367E 00	0.3305E 02	0.4283E 02	1	0.164E 00	0.1665E-01	0.9404E 00	0.9595E 00


2-8
DT